Curriculum Mapping for National Curriculum Statement Grades R-12 and Oracle Academy.

Contents

Executive Summary .................................................................................................................. 3
IT Curriculum Overview ........................................................................................................... 3
  Aims .................................................................................................................................. 3
Oracle Academy Introduction to Computer Science ............................................................ 4
  Introduction to Computer Science Course Overview .......................................................... 4
Mapping Proposal .................................................................................................................. 4
Grade 10 Overview .................................................................................................................. 4
Grade 11 Overview .................................................................................................................. 4
Grade 12 Overview .................................................................................................................. 5
Weighting to Mapping Outline .............................................................................................. 5
Introduction ............................................................................................................................ 6
IT Curriculum Overview ........................................................................................................... 6
  Aims .................................................................................................................................. 6
  Topics .............................................................................................................................. 6
  Resources Required ......................................................................................................... 7
  Approach ........................................................................................................................... 7
  Specific aims of Information Technology ............................................................................ 7
  Time allocation of Information Technology in the curriculum ........................................... 7
  Broad topic layout and progression .................................................................................... 9
  Sub-topic layout and progression for Solution Development ............................................. 10
Oracle Academy Introduction to Computer Science ............................................................ 11
  Introduction to Computer Science Course Overview .......................................................... 12
Mapping Proposal .................................................................................................................. 13
Grade 10 Overview .................................................................................................................. 13
Executive Summary

IT Curriculum Overview

Aims

The National Curriculum Statement Grades R-12 aims to produce learners that are able to:

- identify and solve problems and make decisions using critical and creative thinking
- work effectively as individuals and with others as members of a team organize and manage themselves and their activities responsibly and effectively
- collect, analyze, organize and critically evaluate information
- communicate effectively using visual, symbolic and/or language skills in various modes
- use science and technology effectively and critically showing responsibility towards the environment and the health of others
- demonstrate an understanding of the world as a set of related systems by recognizing that problem solving contexts do not exist in isolation.

The table below provides the six topics and sub-topics to be covered in Information Technology in grades 10-12 and the resources required for teaching IT:

<table>
<thead>
<tr>
<th>Topic Area Sub-Topics</th>
<th>Weighting</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution Development</td>
<td>±60%</td>
<td>Algorithms and Problem Solving</td>
</tr>
<tr>
<td>Communication Technologies</td>
<td>±7%</td>
<td>Networks</td>
</tr>
<tr>
<td>Systems Technologies</td>
<td>±10%</td>
<td>Introduction to Computers</td>
</tr>
<tr>
<td>Internet Technologies</td>
<td>±8%</td>
<td>Internet</td>
</tr>
<tr>
<td>Data and Information Management</td>
<td>±10%</td>
<td>Data Representation</td>
</tr>
<tr>
<td>Social Implications</td>
<td>±5%</td>
<td>Legal Issues</td>
</tr>
</tbody>
</table>

Resources Required

- Computers
- Textbook
- Software
  - Introductory graphical programming language
  - Database Management Software
  - High-level programming language within a visual development environment using an IDE with a GUI builder
  - Internet Browser

The table below provides suggestions for the approximate teaching time and examination time for Information Technology:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Grade 10 Hours</th>
<th>Grade 10 Weeks</th>
<th>Grade 11 Hours</th>
<th>Grade 11 Weeks</th>
<th>Grade 12 Hours</th>
<th>Grade 12 Weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teaching Time: Total</td>
<td>140</td>
<td>35</td>
<td>140</td>
<td>35</td>
<td>100</td>
<td>25</td>
</tr>
<tr>
<td>Examinations</td>
<td>20</td>
<td>5</td>
<td>20</td>
<td>5</td>
<td>48</td>
<td>12</td>
</tr>
</tbody>
</table>
Oracle Academy Introduction to Computer Science

The Oracle Academy, Introduction to Computer Science is designed to provide curriculum and professional development for high school and college teachers who can then provide a thorough foundation in Java programming, database design, SQL programming, and professional skills to students.

Introduction to Computer Science Course Overview

<table>
<thead>
<tr>
<th>Course</th>
<th>Content Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Getting Started with Java Using Alice</td>
<td>8</td>
</tr>
<tr>
<td>Creating Java Programs with Greenfoot</td>
<td>16</td>
</tr>
<tr>
<td>Java Fundamentals</td>
<td>90</td>
</tr>
<tr>
<td>Java Programming</td>
<td>90</td>
</tr>
<tr>
<td>Database Design and Programming with SQL</td>
<td>180</td>
</tr>
<tr>
<td>Programming with PL/SQL</td>
<td>180</td>
</tr>
</tbody>
</table>

Mapping Proposal

Grade 10 Overview
All grade 10 Curriculum Content for Solution Development can be delivered using the course content of Creating Java Programs with Alice and Java Fundamentals. This represents ±60% of the Grade 10 IT Curriculum.

Resources Required

- Computers
- Textbook
  - Online Oracle Academy Curriculum Creating Java Programs with Alice and Java Fundamentals
  - No Additional Textbooks required.
- Software
  - High-level programming language within a visual development environment using an IDE with a GUI builder – Alice 3.1, can be downloaded at no cost.
- Internet Browser

Grade 11 Overview
All grade 11 Curriculum Content for Data and Information Management and Solution Development can be delivered using a subset of the course content of Creating Java Programs with Greenfoot,
Java Fundamentals and Database Design and Programming with SQL. This represents ±70% of the Grade 11 IT Curriculum.

Resources Required

- Computers
- Textbook
  - Online Oracle Academy Curriculum Creating Java Programs with Greenfoot, Java Fundamentals, Database Design and Programming with SQL.
  - No Additional Textbooks required.
- Software
  - Database Management Software – Oracle Application Express (APEX) is a rapid web application development tool for the Oracle database. Provided at no cost.
  - High-level programming language within a visual development environment using an IDE with a GUI builder – Greenfoot, can be downloaded at no cost.
- Internet Browser

Grade 12 Overview

A large percentage of grade 12 Curriculum Content for Data and Information Management (100%) and Solution Development (60%) can be delivered using a subset of the course content of Java Fundamentals, Java Programming, Database Design and Programming in SQL. This represents ±50% of the Grade 12 IT Curriculum.

Resources Required

- Computers
- Textbook
  - Online Oracle Academy Curriculum Java Fundamentals, Java Programming, Database Design and Programming with SQL.
  - No Additional Textbooks required.
- Software
  - Database Management Software – Oracle Application Express (APEX) is a rapid web application development tool for the Oracle database. Provided at no cost.
  - High-level programming language within a visual development environment using an IDE with a GUI builder – Greenfoot / Eclipse, can be downloaded at no cost.
- Internet Browser

Weighting to Mapping Outline

<table>
<thead>
<tr>
<th>Topic Area Sub-Topics</th>
<th>Weighting</th>
<th>% Mapping to Oracle Academy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Grade 10</td>
</tr>
<tr>
<td>Solution Development</td>
<td>±60%</td>
<td>100%</td>
</tr>
<tr>
<td>Introduction to Solution Development</td>
<td></td>
<td>100%</td>
</tr>
<tr>
<td>Application Development</td>
<td></td>
<td>100%</td>
</tr>
<tr>
<td>Software Engineering Principles</td>
<td></td>
<td>50%</td>
</tr>
<tr>
<td>Data and Information Management</td>
<td>±10%</td>
<td>n/a</td>
</tr>
<tr>
<td>Database Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database Design</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Introduction

IT Curriculum Overview

Aims

The National Curriculum Statement Grades R-12 aims to produce learners that are able to:

- identify and solve problems and make decisions using critical and creative thinking
- work effectively as individuals and with others as members of a team organize and manage themselves and their activities responsibly and effectively
- collect, analyze, organize and critically evaluate information
- communicate effectively using visual, symbolic and/or language skills in various modes
- use science and technology effectively and critically showing responsibility towards the environment and the health of others
- demonstrate an understanding of the world as a set of related systems by recognizing that problem solving contexts do not exist in isolation.

Topics

The table below provides the six topics and sub-topics to be covered in Information Technology in grades 10-12 and the resources required for teaching IT:

<table>
<thead>
<tr>
<th>Topic Area</th>
<th>Sub-Topics</th>
<th>Weighting</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution Development</td>
<td></td>
<td>±60%</td>
<td>Algorithms and Problem Solving</td>
</tr>
<tr>
<td></td>
<td>Introduction to Solution Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Application Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Software Engineering Principles</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Communication</td>
<td>Communication Technologies</td>
<td>±7%</td>
<td>Networks</td>
</tr>
<tr>
<td>Technologies</td>
<td>E-communication</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Systems Technologies</td>
<td></td>
<td>±10%</td>
<td>Introduction to Computers</td>
</tr>
<tr>
<td></td>
<td>Hardware</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Software</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Computer Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internet</td>
<td>Internet Technologies</td>
<td>±8%</td>
<td>Internet</td>
</tr>
<tr>
<td></td>
<td>World Wide Web</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet Services</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data and Information Management</td>
<td></td>
<td>±10%</td>
<td>Data Representation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Database Management
Database Design

<table>
<thead>
<tr>
<th>Social Implications</th>
<th>±5%</th>
<th>Legal Issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethical Issues</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Social Issues</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Environmental Issues</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Health Issues</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Computers and Society</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Resources Required**

Computers
Textbook
Software
- Introductory graphical programming language
- Database Management Software
- High-level programming language within a visual development environment using an IDE with a GUI builder
- Internet Browser

**Approach**

The curriculum is designed to introduce learners to the breadth of the field of Information Technology.

**Specific aims of Information Technology**

In Information Technology a learner will:

- use appropriate techniques and procedures to plan solutions and devise algorithms to solve problems using suitable techniques and tools’
- understand and use appropriate communication technologies for information dissemination
- appreciate and comprehend the various systems technologies used in the developing of a computer-based system
- understand that all ICT systems are built upon software engineering principles
- understand and use Internet technologies for various tasks
- comprehend and apply the concepts of data and information management to understand how a knowledge-driven society functions
- understand the social implications of ICTs and how to use ICT technologies responsibly.

**Time allocation of Information Technology in the curriculum**

In Grades 10 and 11 the time allocation for IT is 4 hours per week for 35 weeks. 5 weeks of the school year are taken up by examinations.

The Grade 12 time allocation is 4 hours per week for 28 weeks; 12 weeks of the school year are for examinations.
The table below provides suggestions for the *approximate* teaching time per topic:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Grade 10</th>
<th>Grade 11</th>
<th>Grade 12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hours</td>
<td>Hours</td>
<td>Hours</td>
</tr>
<tr>
<td></td>
<td>Weeks</td>
<td>Weeks</td>
<td>Weeks</td>
</tr>
<tr>
<td>Solution Development</td>
<td>92</td>
<td>90</td>
<td>68</td>
</tr>
<tr>
<td>Communication Technologies</td>
<td>4</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>Systems Technologies</td>
<td>16</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Internet Technologies</td>
<td>14</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>Data and Information Management</td>
<td>8</td>
<td>18</td>
<td>8</td>
</tr>
<tr>
<td>Social Implications</td>
<td>6</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>Teaching Time: Total</td>
<td>140</td>
<td>140</td>
<td>148</td>
</tr>
<tr>
<td>Examinations</td>
<td>20</td>
<td>20</td>
<td>48</td>
</tr>
<tr>
<td>TOTAL :</td>
<td>160</td>
<td>160</td>
<td>148</td>
</tr>
</tbody>
</table>

Basic programming principles and constructs are introduced in Grade 10 through an easy-to-learn, fun tool.

An introductory graphical programming teaching tool such as Scratch/BYOB Scratch is used to introduce learners to important computational skills and concepts, algorithm development, problem solving and programming.

In Grade 11, learners build on the principles and concepts learned in Grade 10 using a high-level programming language that uses an integrated development environment with a GUI builder. Learners are introduced to controls and code and basic object oriented programming (OOP). Event handling principles are reinforced using the form class, attributes, methods and controls.

Skills to manipulate a database through code constructs are also introduced in Grade 12, the principles and constructs are further emphasized through more advanced concepts and problems and learners should be ready to engage with basic structured query language (SQL) code and manipulating a relational database.

The development of computational thinking practices of algorithm development, problem solving and programming underpin solution development and should be emphasized from Grade 10 to Grade 12.

Usability, HCI (human computer interaction) and software engineering principles should be reinforced as part of software development as well as when dealing with websites as part of the Internet Technologies topic.

Algorithmic problem solving in Grade 10 should be dealt with separately at first as an introduction to solution development to develop the learner’s computational thinking practices of algorithm development, problem solving and programming using every day scenarios.

Learners should develop an understanding of the importance of order and precision when developing an algorithm as well as the place of algorithms in software solutions and computing science. Thereafter it should be reinforced, extended and integrated with solution development and programming.

Solution development includes computational thinking and the application of software engineering principles using event driven programming within the object-oriented (OO) paradigm.
Learners should be able to use appropriate practices and tools to:

- solve computational problems through:
  - identifying and analyzing requirements for a specific problem
  - designing effective algorithms
  - converting these to code
  - testing a solution to see if it meets the requirements

- apply the principles of human computer interaction to design functional user interfaces

Broad topic layout and progression
Sub-topic layout and progression for Solution Development

Solution Development

Algorithms and Problem Solving
- Introduction to Algorithms
  - Definition and basic concepts
  - Simple algorithm development
  - Tools to represent an algorithm
  - Tracing and interpreting an algorithm
  - Problem solving steps
- Introduction to Software Engineering Principles
  - Solution development
    - Task definition and analysis
    - Design
    - Implementation
    - Testing
    - Design tools and techniques
    - Simple user interface concepts

Introduction to Solution Development
- Introductory graphical programming tool
  - Development environment, animation characters
  - Nature of objects, their state and behaviour
  - Event handling
  - Interactive Interface
  - Data types
  - Nature and scope of variables
  - Sequence, conditions and iteration
  - Functions (Built-in)
  - Arithmetic operators
  - Relational operators
  - Boolean operators
  - Calculations and operations
  - String operators and manipulation
  - Simple data structures (arrays/lists)
  - Debugging

Application Development
- Introduction to high-level programming language, visual development environment and GUI builder
  - Development environment
  - Event driven principles
  - OOP principles and terminology
  - Applications and projects
  - Basic programming constructs
    - Reinforce Grade 10 concepts
    - Extend Grade 10 concepts
  - Simple validation techniques
  - Basic exception handling
  - Basic solution development
  - Simple database programming – manipulate a single table through code constructs and basic SQL

Databases
- Create, design and modify a relational database
- Import and Export data
- Design and create queries

Extend application development
- Simple user-defined class
- OOP code constructs
- Interactive GUI with database connectivity
- Database transactions utilizing code constructs
- Simple internal and external data structures as part of the solution
- Text based reports
- Validation
- Error handling
- Solution development including data-driven solutions

Structured Query Language (SQL)
- Implement SQL code constructs to perform database transactions
Oracle Academy Introduction to Computer Science

The Oracle Academy, Introduction to Computer Science is designed to provide curriculum and professional development for high school and college teachers who can then provide a thorough foundation in Java programming, database design, SQL programming, and professional skills to students.

Competitive edge in the job market

Students are exposed to technical, business and professional skills that are used in a variety of industries and job roles. Advanced students have the opportunity to pursue Oracle certification – a distinction that provides an additional competitive edge in the job market.

Student-friendly curriculum

This professionally designed curriculum is geared to meet the learning needs of a variety of students, from those interested in gaining broad exposure to business and technical skills to students planning on pursuing a technical education or career. It blends virtual and face-to-face training, hands-on exercises, assessments, and project based learning experiences while leveraging the latest Oracle technologies, allowing teachers and students to easily access the curriculum through a web browser.

Oracle-hosted curriculum and lab environment

Where applicable, Oracle hosts the curriculum and lab environment using state-of-the-art technical infrastructure. The only software setup required is the Java environment and associated software. Thereafter, absolutely no software setup or maintenance is required to deliver the curriculum and its accompanying practices—all you need is a web browser.

Continual Professional Development

The Oracle Academy courses can only be delivered by institutions that have put staff members through the training program offered at no cost by the Oracle Academy.

The content of Introduction to Computer Science courses offered as part of the Oracle Academy curriculum can be delivered as a standalone course or can be delivered as part of a course and mapped to deliver within the IT Curriculum. The following section details which courses have been identified as those containing the IT Curriculum content which can be gained from the Oracle Academy Introduction to Computer Science courses.
Introduction to Computer Science Course Overview

The first year of the Oracle Academy Introduction to Computer Science program comprises a number of courses which are assessed by a combination of restricted response tests and practical exercises. These courses can provide credit transfer for the IT Curriculum. These courses can also help candidates prepare for the Oracle Certified Associate Certification exams. A blended learning approach to the delivery is recommended and students should be encouraged to work through material in their own study time.

Curriculum offered through the Oracle Academy helps students prepare for university-level CS studies and future careers. The Oracle Academy’s Introduction to Computer Science curriculum is designed to awaken student interest in computer science. Recommended for use in secondary schools, technical/vocational schools, and 2-year colleges, it maps to several relevant standards and exams. Our courseware is also modular, enabling faculty to incorporate entire Oracle Academy courses or select lessons to integrate into their existing CS program.

<table>
<thead>
<tr>
<th>Course</th>
<th>Content Hours</th>
<th>Standard &amp; Exam Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Getting Started with Java Using Alice</td>
<td>8</td>
<td>N/A</td>
</tr>
<tr>
<td>Creating Java Programs with Greenfoot</td>
<td>16</td>
<td>N/A</td>
</tr>
<tr>
<td>Java Fundamentals</td>
<td>90</td>
<td>N/A</td>
</tr>
</tbody>
</table>
| Java Programming                            | 90            | Oracle certification¹  
Advanced Placement Computer Science A exam² |
| Database Design and Programming with SQL    | 180           | Oracle certification  
National Workforce Center for Emerging Technology  
National Math, English, and Social Studies  
Ohio IT.WORKS  
State Career Clusters  
IT Diploma (level 3) |
| Programming with PL/SQL                     | 180           | Oracle certification³  
National Workforce Center for Emerging Technology  
National Math, English, and Social Studies  
Ohio IT.WORKS  
State Career Clusters  
IT Diploma (level 3) |

¹ When combined with Java Fundamentals.
² When combined with Java Fundamentals.
³ When combined with Database Design and Programming with SQL.
Mapping Proposal

Grade 10 Overview

All grade 10 Curriculum Content for Solution Development can be delivered using the course content of Creating Java Programs with Alice and Java Fundamentals. This represents ±60% of the Grade 10 IT Curriculum.

Resources Required

- Computers
- Textbook
  - Online Oracle Academy Curriculum Creating Java Programs with Alice and Java Fundamentals
  - No Additional Textbooks required.
- Software
  - High-level programming language within a visual development environment using an IDE with a GUI builder – Alice 3.1, can be downloaded at no cost.
- Internet Browser

Basic programming principles and constructs to be introduced in Grade 10 through the delivery of a subset of Oracle Academy Java Fundamentals and the use of the Alice 3.1 animation and programming tool.

Alice is designed to teach programming theory without the complex language and rules of production languages. Using Alice, you can place objects from a gallery into a virtual world and then create programming statements to animate the objects in the world in a fun and engaging way.

Alice 3.1 will be used to introduce learners to important computational skills and concepts, algorithm development, problem solving and programming.

Alice 3.1 and the Java Fundamentals Curriculum will be used to develop computational thinking practices of algorithm development, problem solving and programming.

Using the Java Fundamentals Curriculum, Algorithmic problem solving in Grade 10 will be dealt with separately at first as an introduction to solution development to develop the learner’s computational thinking practices of algorithm development, problem solving and programming using everyday scenarios.

Learners will use a number of design techniques to develop an understanding of the importance of order and precision when developing an algorithm. Thereafter it will be reinforced, extended and integrated with solution development and programming in the Alice 3.1 environment.

Solution development including computational thinking and the application of software engineering principles will be applied using Alice 3.1 to develop event driven programming within the object-oriented (OO) paradigm.

Learners will be able to use Java Fundamentals and Alice 3.1 to:

- solve computational problems through:
  - identifying and analyzing requirements for a specific problem
  - designing effective algorithms
• converting these to code
• testing a solution to see if it meets the requirements

• apply the principles of human computer interaction to design functional user interfaces

Grade 11 Overview

All grade 11 Curriculum Content for Data and Information Management and Solution Development can be delivered using a subset of the course content of Creating Java Programs with Greenfoot, Java Fundamentals and Database Design and Programming with SQL. This represents ±70% of the Grade 11 IT Curriculum.

Resources Required

• Computers
• Textbook
  o Online Oracle Academy Curriculum Creating Java Programs with Greenfoot, Java Fundamentals, Database Design and Programming with SQL.
  o No Additional Textbooks required.
• Software
  o Database Management Software – Oracle Application Express (APEX) is a rapid web application development tool for the Oracle database. Provided at no cost.
  o High-level programming language within a visual development environment using an IDE with a GUI builder - Greenfoot, can be downloaded at no cost.
• Internet Browser

In Grade 11, learners build on the principles and concepts learned in Grade 10 using the Java Fundamentals Curriculum and implementing solutions in Greenfoot to develop high-level Java programming language solutions.

Greenfoot uses an integrated development environment with a GUI builder. Learners are introduced to controls and code and basic object oriented programming (OOP). Event handling principles are reinforced using classes, attributes, methods and controls.

Greenfoot and the Java Fundamentals Curriculum will be used to develop computational skills and concepts, algorithm development, problem solving and programming.

The Greenfoot development environment promotes Usability, HCI (human computer interaction) and software engineering principles in solution development.

Grade 12 Overview

A large percentage of grade 12 Curriculum Content for Data and Information Management (100%) and Solution Development (60%) can be delivered using a subset of the course content of Java Fundamentals, Java Programming, Database Design and Programming in SQL. This represents ±50% of the Grade 12 IT Curriculum.
Resources Required

- Computers
- Textbook
  - Online Oracle Academy Curriculum Java Fundamentals, Java Programming, Database Design and Programming with SQL.
  - No Additional Textbooks required.
- Software
  - Database Management Software – Oracle Application Express (APEX) is a rapid web application development tool for the Oracle database. Provided at no cost.
  - High-level programming language within a visual development environment using an IDE with a GUI builder – Greenfoot / Eclipse, can be downloaded at no cost.
- Internet Browser

In Grade 12, learners build on the principles and concepts learned in Grade 11 using the Java Fundamentals and Java Programming Curriculum and implementing solutions in either Eclipse or extending Greenfoot to develop high-level Java programming language solutions.

Skills to manipulate a database through code constructs are also introduced in Grade 12 using the Database Programming Curriculum, the principles and constructs are further emphasized through more advanced concepts and problems and learners will be ready to engage with basic structured query language (SQL) code and manipulating a relational database.

Weighting to Mapping Outline

<table>
<thead>
<tr>
<th>Topic Area Sub-Topics</th>
<th>Weighting</th>
<th>Mapping to Oracle Academy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Grade 10</td>
</tr>
<tr>
<td>Solution Development</td>
<td>±60%</td>
<td>100%</td>
</tr>
<tr>
<td>Introduction to Solution Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Engineering Principles</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data and Information Management</td>
<td>±10%</td>
<td>n/a</td>
</tr>
<tr>
<td>Database Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database Design</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Appendix 1 Grade 10 Mapping Detail

<table>
<thead>
<tr>
<th>Algorithm and Problem Solving</th>
<th>Content: Java Fundamentals, Getting Started with Java Using Alice</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction to Algorithms</strong></td>
<td><strong>Introduction to Software Engineering Principles</strong></td>
</tr>
<tr>
<td>- Definitions and basic concepts</td>
<td><strong>Content: Java Fundamentals Section 2, Getting Started with Java Using Alice</strong></td>
</tr>
<tr>
<td>- Simple Algorithm Development</td>
<td><strong>Software Tool: Alice 3.1</strong></td>
</tr>
<tr>
<td>- Tools to represent Algorithm</td>
<td><strong>Solution Development</strong></td>
</tr>
<tr>
<td>- Tracing and Interpreting an Algorithm</td>
<td><strong>Task Definition and Analysis</strong></td>
</tr>
<tr>
<td>- Problem Solving Steps</td>
<td><strong>Design</strong></td>
</tr>
</tbody>
</table>

### Solution Development

- Task Definition and Analysis: JF Section 2 Lesson 11
- Design: JF Section 2 Lesson 11
- Implementation: Java Fundamentals Section 2 Lessons 1 to 11
- Testing: JF Section 2 Lesson 11

### Design Tools and Techniques

- JF Section 2 Lesson 11

### Simple User Interface Concepts

- JF Section 2 Lesson 1 & 2

### Introduction to Software Development

- **Content: Java Fundamentals Section 2, Getting Started with Java Using Alice**
- **Software Tool: Alice 3.1**

### Introduction to Solution Development

- Java Fundamentals Section 2 Lessons 1 to 11

### Introductory Graphical Programming Tool
<table>
<thead>
<tr>
<th>Topic</th>
<th>Section(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development Environment</td>
<td>JF Section 2 Lessons 1 to 2</td>
</tr>
<tr>
<td>Animation Characters</td>
<td>JF Section 2 Lesson 2 to 11</td>
</tr>
<tr>
<td>Nature of Objects, Their State and Behavior</td>
<td>JF Section 2 Lesson 2 to 11</td>
</tr>
<tr>
<td>Event Handling</td>
<td>JF Section 2 Lessons 8 to 9, 11</td>
</tr>
<tr>
<td>Interactive Interface</td>
<td>JF Section 2 Lesson 2 to 11</td>
</tr>
<tr>
<td>Data Types</td>
<td>JF Section 2 Lesson 10</td>
</tr>
<tr>
<td>Nature and Scope of Variables</td>
<td>JF Section 2 Lesson 10</td>
</tr>
<tr>
<td>Sequence, Conditions and Iteration</td>
<td>JF Section 2 Lesson 8</td>
</tr>
<tr>
<td>Functions</td>
<td>JF Section 2 Lesson 7</td>
</tr>
<tr>
<td>Arithmetic Operators</td>
<td>JF Section 2 Lesson 9</td>
</tr>
<tr>
<td>Relational Operators</td>
<td>JF Section 2 Lesson 9</td>
</tr>
<tr>
<td>Boolean Operators</td>
<td>JF Section 2 Lesson 6, 8, 10</td>
</tr>
<tr>
<td>Calculations and Operations</td>
<td>JF Section 2 Lesson 4, 7, 9, 10</td>
</tr>
<tr>
<td>String Operators and Manipulation</td>
<td>JF Section 2 Lesson 4</td>
</tr>
<tr>
<td>Simple Data Structures (Arrays, Lists)</td>
<td>JF Section 2 Lesson 10</td>
</tr>
<tr>
<td>Debugging</td>
<td>JF Section 2 Lesson 2 to 11</td>
</tr>
</tbody>
</table>
### Algorithm and Problem Solving

**Content**: Java Fundamentals, Creating Java Programs with Greenfoot, Database Design and Programming with SQL

**Software Tool**: Greenfoot

### Algorithm Design

JF Section 2 Lesson 11

### Solution Development

- Task Definition and Analysis
  - JF Section 3 Lesson 4
- Design
  - JF Section 3 Lesson 4
- Implementation
  - JF Section 3 Lesson 4
- Testing
  - JF Section 3 Lesson 4

### Design Tools and Techniques

JF Section 2 Lesson 11

### Simple User Interface Design

- Usability
  - JF Section 3 Lesson 4
- Functionality
  - JF Section 3 Lesson 4

### Databases

**Content**: Database Design and Programming with SQL

**Software Tool**: Apex

Create, design and modify a relational database

- DD Sections 12, 15 to 17
- DP Sections 1 to 16
<table>
<thead>
<tr>
<th>Import and Export Data</th>
<th>DD Sections 12, 15 to 17</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DP Sections 1 to 16</td>
</tr>
<tr>
<td>Design and Create Queries</td>
<td>DD Sections 12, 15 to 17</td>
</tr>
<tr>
<td></td>
<td>DP Sections 1 to 16</td>
</tr>
<tr>
<td><strong>Introduction to High Level Programming Language</strong></td>
<td><strong>Content</strong>: Java Fundamentals, Creating Java Programs with Greenfoot, Database Design and Programming with SQL</td>
</tr>
<tr>
<td></td>
<td><strong>Software Tool</strong>: Greenfoot</td>
</tr>
<tr>
<td><strong>Use a visual development environment and GUI builder</strong></td>
<td><strong>JF Section 3 Lesson 1</strong></td>
</tr>
<tr>
<td>Development Environment</td>
<td>JF Section 3 Lesson 1</td>
</tr>
<tr>
<td>Event Driven Principles</td>
<td>JF Section 3 Lesson 3</td>
</tr>
<tr>
<td>OOP Principles and terminology</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>Applications and Projects</td>
<td>JF Section 3 Lesson 4, 7, 9</td>
</tr>
<tr>
<td>Basic Programming Constructs</td>
<td></td>
</tr>
<tr>
<td>- Reinforce Grade 10 Concepts</td>
<td>JF Section 3 All</td>
</tr>
<tr>
<td>- Extend Grade 10 Concepts</td>
<td>JF Section 3 All</td>
</tr>
<tr>
<td>Simple Validation Techniques</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>Basic Exception Handling</td>
<td>JF Section 3 Lesson 7</td>
</tr>
<tr>
<td>Basic Solution Development</td>
<td>JF Section 3 All</td>
</tr>
<tr>
<td>Simple Database Programming</td>
<td></td>
</tr>
<tr>
<td>- Manipulate a Single Table</td>
<td>DD Sections 12, 15 to 17</td>
</tr>
<tr>
<td>- Code Constructs</td>
<td>DD Sections 12, 15 to 17</td>
</tr>
<tr>
<td>- Basic SQL</td>
<td>DD Sections 12, 15 to 17</td>
</tr>
</tbody>
</table>
Appendix 3 Grade 12 Mapping Detail

<table>
<thead>
<tr>
<th>Algorithm and Problem Solving</th>
<th>Content : Java Fundamentals / Java Programming, Database Design and Programming with SQL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Software Tool : Greenfoot / Eclipse /</td>
</tr>
<tr>
<td>Algorithm Design</td>
<td>JF Section 2 Lesson 11</td>
</tr>
<tr>
<td>Solution Development</td>
<td></td>
</tr>
<tr>
<td>• Task Definition and Analysis</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>• Design</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>• Implementation</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>• Testing</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>Design Tools and Techniques</td>
<td>JF Section 2 Lesson 11</td>
</tr>
<tr>
<td>Simple User Interface Design</td>
<td></td>
</tr>
<tr>
<td>• Usability</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>• Functionality</td>
<td>JF Section 3 Lesson 4</td>
</tr>
<tr>
<td>Databases</td>
<td>Content : Database Design and Programming with SQL</td>
</tr>
<tr>
<td></td>
<td>Software Tool : Apex</td>
</tr>
<tr>
<td>Create, design and modify a relational database</td>
<td>DD Sections 12, 15 to 17</td>
</tr>
<tr>
<td></td>
<td>DD Sections 15 to 17</td>
</tr>
<tr>
<td>Import and Export Data</td>
<td>DD Sections 12, 15 to 17</td>
</tr>
<tr>
<td>Design and Create Queries</td>
<td>DP Sections 1 to 16</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td></td>
<td>DD Sections 12, 15 to 17</td>
</tr>
<tr>
<td></td>
<td>DP Sections 1 to 16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Introduction to High Level Programming Language</th>
<th>Content: Java Fundamentals, Database Design and Programming with SQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software Tool: Greenfoot / Eclipse / Apex</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Extend Application Development</th>
<th>JP Sections 1 to 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple User Defined Class</td>
<td>JP Sections 2 to 3</td>
</tr>
<tr>
<td>OOP Code Constructs</td>
<td>JP Sections 2 to 3</td>
</tr>
<tr>
<td>Interactive GUI with Database Connectivity</td>
<td></td>
</tr>
<tr>
<td>Database Transactions Utilizing Code Constructs</td>
<td></td>
</tr>
<tr>
<td>Simple Internal and External Data Structures as Part of the Solution</td>
<td></td>
</tr>
<tr>
<td>Text Based Reports</td>
<td></td>
</tr>
<tr>
<td>Validation</td>
<td></td>
</tr>
<tr>
<td>Error Handling</td>
<td></td>
</tr>
<tr>
<td>Solution Development Including Data Driven Solutions</td>
<td></td>
</tr>
<tr>
<td>Structured Query Language SQL</td>
<td></td>
</tr>
<tr>
<td>- Implement SQL code constructs to perform database transactions</td>
<td>DD Sections 12, 15 to 17</td>
</tr>
<tr>
<td></td>
<td>DP Sections 1 to 16</td>
</tr>
</tbody>
</table>
Appendix 4 Getting Started with Java Using Alice

Getting Started with Java Using Alice

Lesson 1: Prepare for this Course
Lesson 2: Get Started with Alice 3
Lesson 3: Add and Position Objects
Lesson 4: Use Procedures and Arguments
Lesson 5: Add Rotation and Randomization
Lesson 6: Declare Procedures
Lesson 7: Use Control Statements
Lesson 8: Use Functions
Lesson 9: Use the IF and WHILE Control Structures
Lesson 10: Use Expressions
Lesson 11: Use Variables
Lesson 12: Use Keyboard Controls
Lesson 13: Develop a Complete Animation
Lesson 14: Animation Design Worksheet and Academic Examples
Appendix 5 Creating Java Programs with Greenfoot

Creating Java Programs with Greenfoot

Lesson 1: Getting Started With Greenfoot
Lesson 2: Using Methods, Variables and Parameters
Lesson 3: Working with Source Code and Documentation
Lesson 4: Developing and Testing an Application
Lesson 5: Using Randomization and Understanding Dot Notation and Constructors
Lesson 6: Defining Methods
Lesson 7: Using Sound and Keyboard Control
Lesson 8: Creating a World, Animating Actors, and Ending a Game
Lesson 9: Understanding Abstraction
Lesson 10: Using Loops, Variables, and Strings
Lesson 11: Putting it All Together with Greenfoot
Appendix 6 Java Fundamentals Course Content

Java Fundamentals - Section 1
Lesson 1: Welcome
Lesson 2: Introduction

Java Fundamentals - Section 2
Lesson 1: Get Started with Alice 3
Lesson 2: Add and Position Objects
Lesson 3: Use Procedures and Arguments
Lesson 4: Add Rotation and Randomization
Lesson 5: Declare Procedures
Lesson 6: Use Control Statements
Lesson 7: Use Functions
Lesson 8: Use the IF and WHILE Control Structures
Lesson 9: Use Expressions
Lesson 10: Use Variables
Lesson 11: Use Keyboard Controls
Lesson 12: Develop a Complete Animation
Lesson 13: Correlating Java Variables, Data Types, and Expressions with Alice 3 Tools
Lesson 14: Correlating Java Methods, Classes, and Other Structures with Alice 3 Tools

Java Fundamentals - Section 3
Lesson 1: Getting Started With Greenfoot
Lesson 2: Using Methods, Variables and Parameters
Lesson 3: Working with Source Code and Documentation
Lesson 4: Developing and Testing an Application
Lesson 5: Using Randomization and Understanding Dot Notation and Constructors
Lesson 6: Defining Methods
Lesson 7: Using Sound and Keyboard Control
Lesson 8: Creating a World, Animating Actors, and Ending a Game
Lesson 9: Understanding Abstraction
Lesson 10: Using Loops, Variables, and Strings
Lesson 11: Putting it All Together with Greenfoot
Lesson 12: Creating an Inventory of Java Fundamentals

Java Fundamentals - Section 4
Lesson 1: Compiling with Eclipse – A First Program
Lesson 2: Using Object Classes and Driver Classes
Lesson 3: Programming with Data Types and Operators
Lesson 4: Using Strings

Java Fundamentals - Section 5
Lesson 1: Using Scanner and Conditional Statements
Lesson 2: Using Program Control Statements

Java Fundamentals - Section 6
Lesson 1: Using Arrays
Lesson 2: Sorting and Searching
Lesson 3: Handling Errors

Java Fundamentals - Section 7
Lesson 1: Creating Classes, Objects, and Methods
Lesson 2: Passing Objects and Overloading Methods
Lesson 3: Understanding Recursion, the Static Modifier, and Nested Classes
Lesson 4: Understanding Inheritance
Lesson 5: Understanding Polymorphism
Appendix 7 Java Programming Course Content

Java Programming - Section 1
Lesson 1: Deploying an Application

Java Programming - Section 2
Lesson 1: Working with Pre-Written Code

Java Programming - Section 3
Lesson 1: Java Class Design
Lesson 1: Generics and Collections
Lesson 1: String Processing
Lesson 1: Exceptions and Assertions
Lesson 1: Input / Output Fundamentals

Java Programming - Section 4
Lesson 1: Creating a Final Project
Appendix 8 Database Design Course Content

This section details the contents offered by part one of Introduction to Computer Science and Business: Database Design.

**Database Design - Section 1**
Lesson 1: Introduction to the Oracle Academy
Lesson 2: Data vs. Information
Lesson 3: History of the Database
Lesson 4: Major Transformations in Computing

**Database Design - Section 2**
Lesson 1: Conceptual & Physical Models
Lesson 2: Entities, Instances, Attributes and Identifiers
Lesson 3: Entity Relationship Modelling and ERDs

**Database Design - Section 3**
Lesson 1: Identifying Relationships
Lesson 2: ER Diagramming Conventions
Lesson 3: Speaking ERDish and Drawing Relationships
Lesson 4: Matrix Diagrams

**Database Design - Section 4**
Lesson 1: Supertypes and Subtypes
Lesson 2: Documenting Business Rules

**Database Design - Section 5**
Lesson 1: Relationship Transferability
Lesson 2: Relationship Types
Lesson 3: Resolving Many-to-Many Relationships
Lesson 4: Understanding CRUD Requirements

**Database Design - Section 6**
Lesson 1: Artificial, Composite and Secondary UID
Lesson 2: Normalization and First Normal Form
Lesson 3: Second Normal Form
Lesson 4: Third Normal Form

**Database Design - Section 7**
Lesson 1: Arcs
Lesson 2: Hierarchies and Recursive Relationships
Lesson 3: Modeling Historical Data

**Database Design - Section 8**
Lesson 1: Presentation of the ERD to the client

**Database Design - Section 9**
Lesson 1: Modeling Change: Time
Lesson 2: Modeling Change: Price
Lesson 3: Adding the Time Element an ERD

**Database Design - Section 10**
Lesson 1: Drawing Conventions for Readability
Lesson 2: Generic Modeling

**Database Design - Section 11**
Lesson 1: Introduction to Relational Database Concepts
Lesson 2: Basic Mapping: The Transformation Process
Lesson 3: Relationship Mapping
Lesson 4: Subtype Mapping

Database Design - Section 12
Lesson 1: Introduction to Oracle Application Express
Lesson 2: SQL Introduction: Querying the Database
Lesson 3: Basic Table Modifications
Lesson 4: System Development Life Cycle

Database Design - Section 13
Lesson 1: Project Overview and Getting Started
Lesson 2: Presentation Project Management
Lesson 3: Final Presentation Components

Database Design - Section 14
Lesson 1: Creating Tables for the Final Presentation
Lesson 2: Preparing Written Documentation
Lesson 3: Preparing Visual Materials
Lesson 4: Final Presentations

Database Design - Section 15
Lesson 1: Anatomy of a SQL Statement
Lesson 2: Oracle Database Environment
Lesson 3: Using Applications
Lesson 4: Relational Database Technology
Database Design - Section 16
Lesson 1: Working with Columns, Characters, and Rows
Lesson 2: Limit Rows Selected
Lesson 3: Comparison Operators

Database Design - Section 17
Lesson 1: Logical Comparisons and Precedence Rules
Lesson 2: Sorting Rows
Lesson 3: Introduction to Functions – Single Row Functions

Database Design – Appendix A
Lesson 1: What's in Your Future?
Lesson 2: What is a Consultant?
Lesson 3: Speaking in Public
Lesson 4: Leaders in Information Technology
Lesson 5: Creating a Career Portfolio
Lesson 6: Interests, Skills, and Achievements
Lesson 7: Work Experience and Community Involvement
Lesson 8: Creating a Resume
Appendix 9 Database Programming with SQL

This section details the contents offered by part one of Introduction to Computer Science and Business: Database Design and Programming with SQL.

Database Programming with SQL - Section 1
Lesson 1: Case and Character Manipulation
Lesson 2: Number Functions
Lesson 3: Date Functions

Database Programming with SQL - Section 2
Lesson 1: Conversion Functions
Lesson 2: NULL Functions
Lesson 3: Conditional Expressions

Database Programming with SQL - Section 3
Lesson 1: Cross Joins and Natural Joins
Lesson 2: Join Clauses
Lesson 3: Inner versus Outer Joins
Lesson 4: Self-Joins and Hierarchical Queries

Database Programming with SQL - Section 4
Lesson 1: Review of Joins
Lesson 2: Group Functions
Lesson 3: COUNT, DISTINCT, NVL

Database Programming with SQL - Section 5
Lesson 1: Using GROUP BY and HAVING Clauses
Lesson 2: Using ROLLUP and CUBE Operations, and GROUPING SETS

Lesson 3: Using SET Operators

**Database Programming with SQL - Section 6**

Lesson 1: Fundamentals of Subqueries
Lesson 2: Single-Row Subqueries
Lesson 3: Multiple-Row Subqueries
Lesson 4: Correlated Subqueries

**Database Programming with SQL - Section 7**

Lesson 1: Insert Statements
Lesson 2: Updating Column Values and Deleting Rows
Lesson 3: DEFAULT Values, MERGE, and Multi-Table Inserts

**Database Programming with SQL - Section 8**

Lesson 1: Creating Tables
Lesson 2: Using Data Types
Lesson 3: Modifying a Table

**Database Programming with SQL - Section 9**

Lesson 1: Ensuring Quality Query Results

**Database Programming with SQL - Section 10**

Lesson 1: Defining NOT NULL and UNIQUE Constraints
Lesson 2: PRIMARY KEY, FOREIGN KEY, and CHECK Constraints
Lesson 3: Managing Constraints

**Database Programming with SQL - Section 11**
Lesson 1: Creating Views
Lesson 2: DML Operations and Views
Lesson 3: Managing Views

Database Programming with SQL - Section 12
Lesson 1: Working with Sequences
Lesson 2: Indexes and Synonyms

Database Programming with SQL - Section 13
Lesson 1: Controlling User Access
Lesson 2: Creating and Revoking Object Privileges
Lesson 3: Regular Expressions

Database Programming with SQL - Section 14
Lesson 1: Database Transactions

Database Programming with SQL - Section 15
Lesson 1: Cartesian Products and the Join Operation
Lesson 2: Nonequijoins
Lesson 3: Outer Joins

Database Programming with SQL - Section 16
Lesson 1: Testing
Lesson 2: Final Project Database Creation
Lesson 3: Final Exam Review
Database Programming with SQL - Section 17
Lesson 1: Ensuring Quality Query Results – Advanced Techniques

Database Programming with SQL – Appendix A
Lesson 1: The Changing Nature of the Job Market
Lesson 2: Searching for a Job
Lesson 3: Written Communication
Lesson 4: Interviewing
Lesson 5: Networking