# Level 1

## AS92005: Outcome 5 Credits

| **Achieved**  **Develop a digital technologies outcome** | **Comments** |  |
| --- | --- | --- |
| Identifying the purpose, potential users, requirements, and specifications of the outcome | Complete section 1 correctly |  |
| Using appropriate tools and techniques of a digital technologies domain to produce an outcome that addresses the requirements and specifications | Use Python and SQLite to create a program to interface with a database that functions as expected and hands it in on time.  Application has at least read functionality and can output data from the database from users input.  Handed in all project code as a github link or as zip file.  May not have used github  Final project may not look exactly like design and might not have all the intended functionality. |  |
| Testing the outcome to ensure basic functionality. | Teacher observation of testing and the application looks and functions mostly as expected.  May not have done sql query table. |  |
| **Merit**  **Refine a digital technologies outcome** |  |  |
| Following relevant conventions of a digital technologies domain | Meet at least 4 of the following (teacher discretion)  Conventions include but are not limited to:   * Using an organised file structure for your project files * Using well named files and folders in your project * Has appropriate function names that describe the purpose * Using code comments where appropriate * Following common language conventions for Python and SQL * Following Database Design standards (eg normalized data to 3NF- well structured database) * Using version control including descriptive commit messages |  |
| Using information from testing to make improvements to the outcome's fitness for purpose. | Did testing during development. This can be observed or exemplified through a git commit log or multiple versions of code with incremental improvements. |  |
| **Excellence**  **Enhance a digital technologies outcome** |  |  |
| using information from trialling the outcome with others to improve its fitness for purpose | Lots of improvement through at least three versions. Including relevant feedback from others and implementing suggested changes where appropriate.  Teachers should set “Checkpoints” with the whole class to help facilitate this. |  |
| applying tools and techniques effectively in the production of a fit-for-purpose outcome. | The final outcome meets the design and specifications and shows an excellent grasp of Python and SQL to create an above average program that follows all conventions.  Adds features not included in the tutorial like Login/admin/passwords etc. |  |

# 

# Level 2

## AS91897- Use advanced techniques to develop a database

**Credits:** 4 (internal)

**NZQA:**  <https://www.nzqa.govt.nz/nqfdocs/ncea-resource/achievements/2019/as91892.pdf>

| **Achieved- develop a database** | **Evidence** | ✔ |
| --- | --- | --- |
| designing the structure of the data |  |  |
| using appropriate tools and advanced techniques to organise, query and present data for a purpose and end users |  |  |
| applying appropriate data integrity and testing procedures |  |  |
| explaining relevant implications. |  |  |
| **Merit- Develop and informed database** |  |  |
| using information from testing procedures to improve the quality of the database |  |  |
| structuring, organising and querying the data logically |  |  |
| addressing relevant implications. |  |  |
| **Excellence- develop a refined database** |  |  |
| iterative improvement throughout the design, development and testing process |  |  |
| presenting the data effectively for the purpose and end users. |  |  |

# Level 3

## AS91902- Use complex techniques to develop a database

**Credits:** 4 (internal)

**NZQA:**  <https://www.nzqa.govt.nz/nqfdocs/ncea-resource/achievements/2019/as91902.pdf>

| **Achieved- develop a database** | **Comments** | ✔ |
| --- | --- | --- |
| designing the structure of the data |  |  |
| using appropriate tools and appropriate techniques to organise, query and present data for a purpose and end users |  |  |
| applying appropriate data integrity and testing procedures |  |  |
| addressing relevant implications. |  |  |
| **Merit- Develop and informed database** |  |  |
| using information from testing procedures to improve the quality of the outcome |  |  |
| structuring, organising and querying the data logically |  |  |
| **Excellence- develop a refined database** |  |  |
| iterative improvement throughout the design, development and testing process |  |  |
| using efficient tools and techniques in the outcome’s production |  |  |
| presenting the data effectively for the purpose and end users. |  |  |

# Step Ups

| **Level 2** | **Level 3** |
| --- | --- |
| Advanced techniques will include a selection from:   * linking data in related tables or nodes using queries or keys * writing custom queries to filter and/or sort data * using logical, mathematical and/or wildcard operators * customising presentation of the data * using custom forms to add user input to the database * setting validation rules for data entry. | Complex techniques will include a selection from:   * structuring the data using multiple tables or nodes * creating queries which insert, update or delete to modify data * creating customised data displays from multiple tables or nodes (e.g. reports, PDFs, web pages, dashboards, program interfaces) * dynamically linking data between the database and a front-end display * applying data access permissions as appropriate to the outcome. |
| Eg.  Relational database. More than one table.  WHERE, ORDER BY | Eg.  Many to Many  CRUD. ie INSERT, UPDATE, DELETE |

# 

# 

# Programming Standard Checklists

Level 2

## AS91896 - Use advanced programming techniques to develop a computer program

**Level:** 2

**Credits:** 6 (Internal)

**NZQA:**  <https://www.nzqa.govt.nz/nqfdocs/ncea-resource/achievements/2019/as91896.pdf>

| **Achieved** | **Evidence** |  |
| --- | --- | --- |
| writing code for a program that performs a specified task |  |  |
| using advanced techniques in a suitable programming language |  |  |
| setting out the program code clearly and documenting the program with comments |  |  |
| testing and debugging the program to ensure that it works on a sample of expected cases. |  |  |
| **Merit** |  |  |
| documenting the program with appropriate names and comments that describe code function and behaviour |  |  |
| following common conventions for the chosen programming language |  |  |
| testing and debugging the program effectively to ensure that it works on a sample of both expected cases and relevant boundary cases. |  |  |
| **Excellence** |  |  |
| ensuring that the program is a well-structured, logical response to the specified task |  |  |
| making the program flexible and robust |  |  |
| comprehensively testing and debugging the program. |  |  |

## AS91906 - Use complex programming techniques to develop a computer program

**Level:** 3

**Credits:** 6 (Internal)

**NZQA:**  <https://www.nzqa.govt.nz/nqfdocs/ncea-resource/achievements/2019/as91906.pdf>

| **Achieved** | **Evidence** |  |
| --- | --- | --- |
| writing code for a program that performs a specified task |  |  |
| using complex techniques in a suitable programming language |  |  |
| setting out the program code clearly and documenting the program with comments |  |  |
| testing and debugging the program to ensure that it works on a sample of expected cases. |  |  |
| **Merit** |  |  |
| documenting the program with appropriate names and organised comments that describe code function and behaviour |  |  |
| following conventions for the chosen programming language |  |  |
| testing and debugging the program effectively to ensure that it works on a sample of both expected cases and relevant boundary cases. |  |  |
| **Excellence** |  |  |
| ensuring that the program is a well-structured, logical response to the specified task |  |  |
| making the program flexible and robust |  |  |
| comprehensively testing and debugging the program. |  |  |