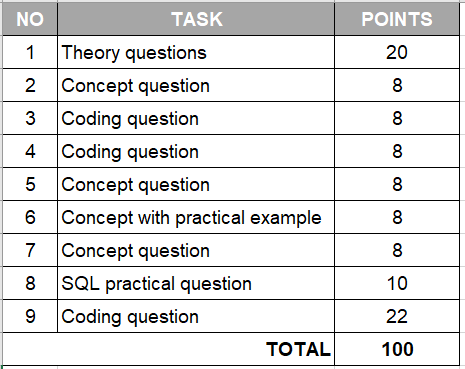
**ASSESSMENT**

Python and MySQL

assessment test 2 hours



|  |  |
| --- | --- |
| 1. **Python theory questions** | **20 points** |

1. What is the program?

A program is a collection of code which preforms an action (gives an output) based on the instructions it is given.

1. What is the process?

A process is a program whilst it is running.

1. What is Cache?

This is part of the memory stored in the CPU

1. What is Thread and Multithreading?

Usually, one thread runs at a time, but you can make it so that multiple threads run at simultaneously. A thread is short for something…

1. What is GIL in Python and how does it work?

The L stands for ‘language’

1. What is Concurrency and Parallelism and what are the differences?

Concurrency is two things happening sat the same time from one source. Whilst parallelism is two things happening at the same time from two different sources.

1. What do these stand for in programming: DRY, KISS, BDUF

DRY: DO RUN YOURSELF – something about doing it yourself

KISS: KEEP IT SUPER SIMPLE – something about simplicity

BDUP:

1. What is Garbage collector? How does it work?

This is where you can chuck things which are no longer useful to you so that it doesn’t waste space in your cloud because cloud space can be expensive. Garbage collector stores it somewhere else so you can potentially get it back.

1. What are ‘deadlock’ and ‘livelock’ in a relational database?

Deadlock might be not letting anyone edit what you’ve done. And livelock maybe letting people view your work but not edit it?

1. What is Flask and what can we use it for?

You install flask in order to use it. I think flask was a package that we pip installed. We either used it to connect python to mySQL or for testing.

|  |  |
| --- | --- |
| 1. **Discuss the difference between Python 2 and Python 3**   Essentially python 2 is the old version. One of the main differences is that in python 2 the print statement doesn’t require brackets. So you can say print “hello world”, but in python 3 you’d have to say print(“hello world”). We use python 3. Another difference is that in python 3 you don’t have to specify anything’s ‘type’ because python 3 can figure this out by itself (I believe this is called being ‘type-oriented’), say if a variable is an integer or a float, but this isn’t the case for python 2. I think that python is generally described as being a ‘dynamic’ language. | **8 points** |

|  |  |
| --- | --- |
| 1. **Write a function that can define whether a word is a Palindrome or not (a word, phrase, or sequence that reads the same backwards as forwards, e.g. *madam*)**   A function would mean that I need an input and I’ll do an if statement, and I think I’ll have to use [::-1], so let’s do the following:  your\_word = str(input(“what is your word?”))  reverse\_your\_word = [your\_word::-1]  If your\_word == reverse\_your\_word:  Print(your\_word)  Else:  Print(“that is not a palindrome”) | **8 points** |

|  |  |
| --- | --- |
| 1. **Write tests for the newly created Palindrome function. Provide a brief explanation for your test case options.**   So I will test that the input is actually a string since the question specified that I needed a ‘word, phrase, or sequence’  IsItAString(test case):  Self.sassertTrue(your\_word == type(string))  IsItNotAString(test case):  Self.assertFalse(your\_word != type(string))  Here’s a test for whether the inputted word is actually a palindrome or not  IsAPanlindrom(test case):  Self.sassertTrue(your\_word == reverse\_your\_word)  IsNotAPanlindrom(test case):  Self.assertFalse(your\_word != reverse\_your\_word) | **8 points** |

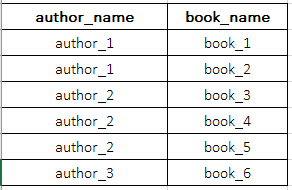
|  |  |
| --- | --- |
| 1. **Agile methodology, Scrum: name at least 3 types of meetings that are exercised by Agile teams and describe the objective of each meeting.**   Daily scrum: this is the 15-minute daily meeting where the team comes together. Each team member discusses three things: what they will do today, what they did yesterday, any problems they faced. The Scrum master can then support the team accordingly.  Scrum retrospective: this is when you look back at all you have done as a team. Usually in the form of a presentation to all stakeholders. You can test (this is called a ‘demo’) the program/app and get feedback/ make changes afterwards.  The initial meeting: this is where you are presented with the project and the multidisciplinary team must organise themselves.You are briefed on all the requirements from all the stakeholders. | **8 points** |

|  |  |
| --- | --- |
| 1. **Exception handling in Python, explain what each of the following blocks means in the program flow:**   **Try, except, else, finally**  Try: try and see if the following things works or raises an exception  Except: this tells you want to do if a particular exception is raised, and is like a caveat for that particular exception  Else: do this if the exception above is not raised  Finally: this always runs | **8 points** |

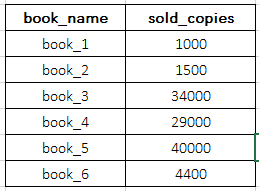
|  |  |
| --- | --- |
| 1. **How can we connect a Python program (process) with a database? Explain how it works and how do we fetch / insert data into DB tables from a python program.**   You need to pip install mysqlconnector I believe. Then you need to remember the password for your workbench. If you have a separate .py file called config (for example) you can put your USERNAME, HOST, PASSWORD in there. Then open up a new .py file and import from config your USERNAME, HOST, PASSWORD (this must be the PASSWORD that you created when you first downloaded workbench).  The next thing you do is import things and I think we used something that began with an @ symbol. There are certain key words that are necessary like FETCH and INSERT.  If you want to change something in the workbench through Python you don’t need to use the semi colon in Python like we do in workbench. Apart from that though you can use normal mySQL syntax. | **8 points** |

|  |  |
| --- | --- |
| 1. **Given two SQL tables below: authors and books.**  * **The authors dataset has 1M+ rows** * **The books dataset also has 1M+ rows**   Create an SQL query that shows the TOP 3 authors who sold the most books in total!  My answer:  What do I need to do? Select the most copies sold, from the book\_name, order them by the author\_name. I think I should do a JOIN based on the book\_name column.  SELECT sold\_copies a , book\_name a, author\_name b,  JOIN book\_name a = book\_name b  WHERE sold\_copies > 29000  ORDER BY book\_name  RETURN author\_name b | **10 points** |

**AUTHORS**

****

**BOOKS**

****

|  |  |
| --- | --- |
| 1. **TWO NUMBER SUM:**  * Write a function that takes in a non-empty array of distinct integers and an integer representing a target sum. If any two numbers in the input array sum up to the target sum, the function should return them in an array, in any order. If no to numbers sum up to the target sum, the function should return an empty array. * Note that the target sum has to be obtained by summing two different integers in the array. You cannot add a single integer to itself in order to obtain the target sum. * You can assume that there will be at most one pair of numbers summing up to the target sum.   **Sample Input:** numbers = [3, 5, -4 ,8, 11, 1, -1, 6] target\_sum = 10  **Sample Output:** [-1, 11] the numbers can be in any order, it does not matter.  My answer:  Import random  My\_numbers = [3, 5, -4 ,8, 11, 1, -1, 6]  target\_sum = 10  if sum(random.sample[my\_numbers,2]) == 10:  print(random.sample[my\_numbers,2])  else:  print([]) | **22 points** |