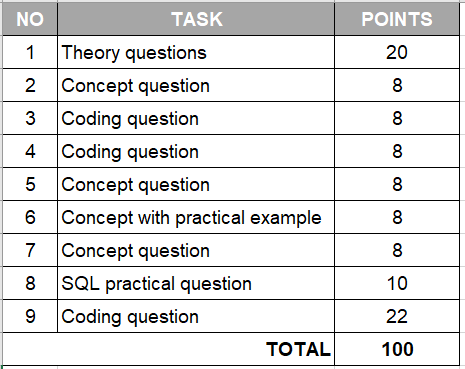
**ASSESSMENT**

Python and MySQL

assessment test 2 hours



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

1. What is the program?

Set of instructions written in a sequential manner to solve a task

1. What is the process?

Process is an instance of a program executed by more than one threads

1. What is Cache?

It’s a software/hardware memory component with a data storing technique used to store some variables, data to be used further in the program. Cache improved speed of execution

1. What is Thread and Multithreading?

Thread: small unit like an individual part of a program that can be executed independently

Multi-threading: Phenomenon and capability of executing more than one thread at once using same resources

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

GIL: Global Interpreter Lock. It is a phenomenon in which only one thread has control over interpreter at particular moment of time. This eliminates race-out condition.

Working: GIL uses concept called ticks and checks the thread for every 100 ticks

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

Concurrency: Making progress on more than one task but not simultaneously.

Parallelism: Splitting task to smaller subtasks and executing on multiple CPU all at same time.

Diff:

In concurrency tasks are not executed once , but changes are allowed while in parallelism tasks are executed at same time.

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

DRY: Don’t Repeat Yourself

KISS: Keep It Simple, Stupid

BDUF: Bring Design Up Front

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

Garbage collector: It is something like a bin to dump un referenced, deleted python objects

Working:

When an object has zero references assigned to it at any particular time, then memory assigned to it will be de-allocated and the object will be deleted.

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

Deadlock: Condition when two or more database tasks are waiting for eachother to use mutually sharable resources and none of the task is willing to give resources.

Livelock:

Situation when two or more process keep changing their status by following the changes in other and this prevents them from finishing their tasks.

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

It is a web framework providing libraries to build light weighted web application in python language.

|  |  |
| --- | --- |
| 1. **Discuss the difference between Python 2 and Python 3** | **8 points** |

1.Storing of Unicode: In python3, Unicode is stored as “String” while in python2, Unicode should be specified with -u

2. Range() function: Python3 has iterable range() function , while python2 has xrange() function

3. range function return: In python3 , range() function when iterated result list of individual values while in python2 xrange() function returns values as an ‘object’

4. Exceptions:

In python3, exceptions are written in ‘paranthesis’ , while in python2, exceptions are written in ‘notations’.

5. Easier syntax: Python3 has well represented easier, simple syntax compared to python2

|  |  |
| --- | --- |
| 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*)** | **8 points** |

word=input(“enter string to be checked”)

def paliendrome(word1):

n=len(word1)

for i in range(0,n):

if word1[i]==word1[n-i]:

i=i+1

print(“ paliendrome”)

return true

else:

print(“not palindrome”)

return false

paliendrome(word)

|  |  |
| --- | --- |
| 1. **Write tests for the newly created Palindrome function. Provide a brief explanation for your test case options.** | **8 points** |

import unittest

from folder import paliendrome.py

class test(Testcase):

def test(self):

result=paliendrome(“madam”)

expected= true

self.assertEqual(result, expected)

return true

Explaination: Import unittest module,

from folder of paliendrome file import paliendrome.py file which has function and code to be tested

define a class named test and inherit TestCase class from unittest module

define a function and pass “self” object

pass the string to be tested into paliendrome function(imported from paliendrome.py)

This call paliendrome function and passes argument to it and return result to given result variable

Define expected result and check this expected result with “assertEqual” function and if it is paliendrome this return “true”

Repeat this case for few other strings and test

|  |  |
| --- | --- |
| 1. **Agile methodology, Scrum: name at least 3 types of meetings that are exercised by Agile teams and describe the objective of each meeting.** | **8 points** |

**1. Sprint Planning**: Meeting held at the beginning of every sprint. Main goal is plan the implementation of new sprint.

**Objectives:**

Selecting items from product backlog and designing plan to finish them.

Select any other tools and procedures to be followed newly or updating any procedures for effective task completion for that sprint.

**2. Daily Scrum**: It is simple meet conducted everyday at a maximum for 15 minutes mostly before starting work to achieve sprint goals

**Objectives:**

The main objective is to review previous day work and plan for that current day.

Select and assign the tasks from product backlog to be finished.

Simply “what the progress on yesterday, what to be done today.

**3. Sprint Retrospective**: It is to plan wats to increase quality and effectiveness

**Objectives:**

What went well and what to be improved further

How to improve next sprint , what to implement

Find the best past improvements to be followed for next sprint.

|  |  |
| --- | --- |
| 1. **Exception handling in Python, explain what each of the following blocks means in the program flow:**   Try, except, else, finally | **8 points** |

Try: This block has the condition and code to be executed when condition satisfied

Except: The block has exception to be raised when any error occurs in try block

(specifying what the error and type)

Else: this block has the code to be executed when condition not met in try block and is written inside try block( block executes when no exception raised in try block0

Finally: This block has to code to be executed definitely whatever happens in try -except block(Compulsory condition for both try, except conditions)

|  |  |
| --- | --- |
| 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.** | **8 points** |

**Command: pip3 install mysql\_connector on terminal**

**Step1: Create config.py file**

**HOST=host**

**USER: “root”**

**PASSWORD:”password”**

**All these credentials should match database credentials**

**Step2: Utils.py**

**This file has all the necessary functions to be used**

**Connection.py: File to setup and manage connections with database**

**This file has cursor initialised**

**main.py: This file has all the code to be worked on and it has code to connect to remaining files**

**How connected to database:**

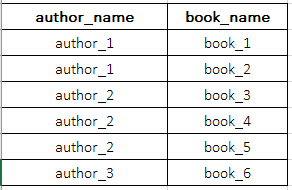
**The main file has all the remaining files imported and when run, sequentially remaining files will be run in accordance to the code.**

**How to fetch data:**

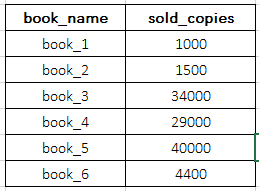
* + - 1. **Connect to database and initialise cursor in connection.py**
      2. **In main.py, connect cursor to database, and open**
      3. **Define function to fetch/insert with cursor called**
      4. **Write query to fetch/insert data from database in query block and call the function**
      5. **Finally close cursor after function and task executed.**

|  |  |
| --- | --- |
| 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! | **10 points** |

**AUTHORS**

****

**BOOKS**

****

Select j.author\_name, j.book\_name from

(select a.author\_name, a.book\_name, b.sold\_copies

from Authors a

inner join Books b

on a.book\_name=b.book\_name) as j

group by j.author\_name

order by sum(sold\_copies) desc

having rownum<=3;

|  |  |
| --- | --- |
| 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. | **22 points** |

num\_array=list()

num=input(“Enter how many element you want)

print(“Enter numbers in array:”)

for i in range(int(num)):

n=input(“num:”)

num\_array.append(int(n))

print(‘Array:”, num\_array)

sum=int(input(“please enter sum”))

def my\_function(arr,sum):

temp=0

for i in range(len(arr)):

for j in range(i+1, len(arr)):

temp=arr[i]+arr[j]

if temp==sum:

print(“my\_numbers=[{},{}]”.format(arr[i], arr[j]))

print(“target\_sum=”’{}’.format(sum))

#else :

#Print(“[]”) (printing empty list)

my\_function(num\_array, sum)