**FSDS MAY BATCH 2022(Python Assignment -7)**

**Submitted by: Shubham Tiwari**

Q1: What is the purpose of the try statement?

Ans: Basically the Python try…except statement runs the code under the “try” statement. If this code does not execute successfully, the program will stop at the line that caused the error and the “except” code will run.In detail it works in the following manner.

* First, the **try** clause is executed i.e. the code between **try** and **except** clause.
* If there is no exception, then only the **try** clause will run, **except** the clause is finished.
* If any exception occurs, the **try** clause will be skipped and **except** clause will run.
* If any exception occurs, but the **except** clause within the code doesn’t handle it, it is passed on to the outer **try** statements. If the exception is left unhandled, then the execution stops.
* A **try** statement can have more than one **except** clause.

Q2: What are the two most popular try statement variations?

Ans: The two most popular try statement variations can be:

* We can use the else clause on the try-except block which must be present after all the except clauses. The code enters the else block only if the try clause does not raise an exception.
* Secondly, there is try-except condition: The try block is used to check some code for errors i.e the code inside the try block will execute when there is no error in the program. Whereas the code inside the except block will execute whenever the program encounters some error in the preceding try block.

Q3: What is the purpose of the raise statement?

Ans: **“Raise” Keyword** is used to raise exceptions or errors. The raise keyword raises an error and stops the control flow of the program. It is used to bring up the current exception in an exception handler so that it can be handled further up the call stack. The syntax for raise statement is as given below:

*raise  {name\_of\_ the\_ exception\_class}*

Q4: What does the assert statement do, and what other statement is it like?

Ans: The assert statement in simpler terms, we can say that assertion is the boolean expression that checks if the statement is True or False. If the statement is true then it does nothing and continues the execution, but if the statement is False then it stops the execution of the program and gives an error.This statement takes as input a boolean condition, which when returns true doesn’t do anything and continues the normal flow of execution, but if it is computed to be false, then it raises an Assertion Error along with the optional message provided. The basic syntax is as given below:

***Syntax :****assert condition, error\_message(optional)*

It looks similar just like raise statement .

Q5: What is the purpose of the with/as argument, and what other statement is it like?

Ans: “**With statement”** is used in exception handling to make the code cleaner and much more readable. It simplifies the management of common resources like file streams. The “with” statement is a replacement for commonly used try/finally error-handling statements. A common example of using the “with” statement is opening a file.

To open and write to a file in Python, we can use the with statement as follows:

with open("example.txt", "w") as file:

file.write("Hello World!")

The “with” statement automatically closes the file after we have completed writing it.

It looks similar to try/finally error-handling statements.