Q1. Describe three applications for exception processing.

**Ans. Three applications for exception processing :**

* **Exception handling allows you to separate error-handling code from normal code, exceptions helps you to remind yourself of what the program expects.**
* **It clarifies the code and enhances readability and allows you to stimulate consequences as the error-handling takes place at one place and in one manner.**
* **Raising an exception helps you to break the current code execution and returns the exception back to expection until it is handled, processing exceptions for components which can't handle them directly.**

Q2. What happens if you don't do something extra to treat an exception?

**Ans. The runtime system will abort the program (i.e. crash) and an exception message will print to the console.**

Q3. What are your options for recovering from an exception in your script?

**Ans. try/except : Catch the error and recover from exceptions hoist by programmers or Python itself.**

**try/finally : Whether exception occurs or not, it automatically performs the clean-up action.**

Q4. Describe two methods for triggering exceptions in your script.

**Ans**. **Assert : Triggers an exception conditionally in the code.**

**Raise : Manually triggers an exception in the code.**

Q5. Identify two methods for specifying actions to be executed at termination time, regardless of whether or not an exception exists..

**Ans. Finally clause and else clause are two methods which are used for this purpose.**