1. True or False – The main reason Automation Testing will be conducted is to make sure that humans have not made a mistake while doing manual testing.
   False. We automate the manual test cases so we can conduct our testing in a more efficient manner. Through automation, we are able to achieve increased test coverage while reducing costs and time required for manual execution.

2. What is the purpose of Output Values in UFT?
   The purpose of an Output Value is to capture values from the AUT so that they can be used later in the test for various reasons.

3. What is the purpose of the Object Repository in UFT?
   The object repository is used to store information about the objects that UFT has learned. The relevant information stored in the OR consists of the Objects and their description properties and values. UFT uses the objects' properties and associated values to identify objects during run-time. The properties that UFT 'learns' about an object are determined based on the settings set in the 'Object Identification' window for each object class.

4. What are the 3 default add-ins that come with UFT?
   1. ActiveX
   2. Visual Basic
   3. Web
   Windows functionality is default within UFT – i.e. UFT can test Windows applications without the need for any Add-in.

5. Explain all the steps necessary to create a script using the Record and Run method.
   1. Create a New GUI Test in UFT with the Web Add-in selected
   2. Record → Record and Run Settings

3. Under the 'Web' tab, select the 2nd radio button to 'Open the following address when a record or run session begins:'
4. Enter the URL to be tested
5. Select the browser
6. Click on 'Apply' button and close the window
7. Click on the 'Record' button and begin recording the test by performing the actions on the application
8. Add Checkpoints and Output values as needed to enhance the test

6. Why is automation testing necessary? What is the purpose of Automating a manual test case?
   We automate the manual test cases so we can conduct our testing in a more efficient manner. Through automation, we are able to achieve increased test coverage while reducing costs and time required for manual execution.

   One of the main reasons for automation testing is to perform regression testing. Regression testing is done to verify that changes to an application have not broken those areas of the application that were previously working properly. Since regression testing requires us to execute the same tests over and over again, it's a good idea to automate those tests to reduce the workload and make testing more efficient.

   We will create an automation test (automate a manual test case) that we know will pass when executed against the current version of the application. However, we will formally execute the automation test when a new version of the application is released. The idea is to verify that current functionality is still working given the change.

7. What does a Standard Output Value do exactly?
   A Standard Output Value will capture property value(s) from an identified object sitting in the Application Under Test (AUT) during run time. It will then store the captured property value(s) in either the Data Table or in Environment Variables.

   The purpose of an Output Value is to capture values from the AUT so that they can be used later in the test for various reasons.

8. What are the different types of Checkpoints available in UFT? List **all** of them, and explain 3 of them.
   1. **Standard Checkpoint**: Compares the expected values of object properties captured during recording to the object's current values during a run session
   2. **Page Checkpoint**: A Standard Checkpoint created for a web page can be called a Page Checkpoint. It is used to check total number of links & images on a web page. Page Checkpoints can be used to check Load Time i.e. time taken to load a web page.
   3. **Bitmap Checkpoint**: Helps a user in checking the bitmap of an image or a full web page. It does a pixel by pixel comparison between actual and expected images.
   4. **Text Checkpoint:** Used to check expected text in a web-page or application. This text could be from a specific region of the application or a small portion of text displayed
   5. **Accessibility Checkpoint:** Verifies compliance with World Wide Web Consortium (W3C) instructions and guidelines for Web-based technology and information systems. These Guidelines make it easy for disabled to access the web. (also look at *Section 508 Compliance Testing*)
   6. **Database Checkpoint:** Create a query during record time and database values are stored as expected values. The same query is executed during run time and actual & expected values are compared.
   7. **Table Checkpoint:** Check the contents of cells of a table (grid) appearing in your environment. You can also check various table properties like row height , cell width and so on. Table Checkpoint is similar to *Database Checkpoint*
   8. **XML Checkpoint:** Verify XML Data, XML Schema, XML Data

9. What is the difference between Mandatory Properties, Assistive Properties, and Ordinal Identifiers? Explain all three.
   **Mandatory Properties**: Those properties that UFT MUST learn in order to identify an object.
   **Assistive Properties:** Those properties that UFT MAY learn in order to identify an object *IF* the Mandatory Properties were not enough to uniquely identify the object.
   **Ordinal Identifiers**: Each object in the application is given an ordered number which identifies the object in a specified numbered ordered.

Example: 1st object, 2nd object, 3rd object, etc.

There are 3 types of Ordinal Identifiers: Index, Location, Creation Time

All ordinal identifier values start at 0

Ex.: 1st Object → Index Value = 0

2nd Object → Index Value = 1

3rd Object → Index Value = 2…and so on…

10. Explain the differences and similarities between the 2 methods for creating automation scripts that we have learned so far.

We have learned 2 methods of designing automation tests till now: Record and Run and Drag-and-Drop

Record and Run
- Set the Record and Run settings
- Hit Record
- Record the steps to be automated
- UFT will create the script
- UFT will create the Object Repository
- Enhance our script using Checkpoints and Output Values

Drag-and-Drop
- Test Engineer will Build the Object Repository
- Test Engineer will Design the Script
- Test Engineer will enhance the script using Checkpoint and Output Values

11. Currently the object repository learns the following properties for a Link object:
    a. Innertext
    b. Name
    c. Html Tag

    Write the steps necessary to have the 'href' property (and associated value) also stored in the object repository ***if*** UFT needs to learn it to uniquely identify a Link object.

    1. Tools → Object Identification
    2. Select 'Web' under the Environment section
    3. Click on 'Link' under Object Class
    4. Under the 'Assistive Properties' section, click on the

'Add/Remove' button
5. Select 'href' from the list of properties and click 'OK'
6. Now href is added under the Assistive Properties section

12. Write the steps necessary to create a Shared Object Repository and use the SOR to design an automation test in UFT.

**Steps to Create a SOR:**

1. Build a Local Object Repository
2. In the Object Repository, Click on File → Export Local Objects
3. Create a New Folder and call it 'Shared Object Repositories'
4. Within the newly created 'Shared Object Repositories' folder, save the OR file by giving it a name and clicking on 'Create'
5. In the Windows File Explorer, navigate to the 'Shared Object Repositories' folder and verify that the .tsr Shared Object Repository file has been created

**Associate a SOR with a test:**

1. Create a Shared Object Repository
2. In UFT, click on Resources → Associate Repositories
3. Click on the Green '+' button
4. Locate the .tsr file and click on 'Open'
5. Under the 'Available Actions' window, select 'Action 1'
6. Click on the ' > ' button and bring 'Action 1' under the 'Associated Actions' window
7. Click 'OK'
8. Open the Object Repository and verify that the SOR has been associated successfully

**Create a test using the object in the SOR via the Drag and Drop Method:**

1. After Associating the SOR with the test, launch the Object Repository
2. Use the Drag-and-Drop method to create the automation script
3. Incorporate Checkpoints and Output Values by starting 'record mode' as needed throughout the script.

13. What does the '***SetSecure***' method do? Why is this important when creating automation scripts?
***Bonus Question – 15 points extra credit.**
***You must answer <u>both</u> parts of this question <u>completely</u> and <u>*in your own words*</u> to receive credit**
The SetSecure method in UFT decrypts text that has already been encrypted.

The SetSecure method is an important feature of UFT because it allows for sensitive data to be passed from user to user among Test Engineering teams without violating privacy policies.

Example:
If we need to automate the login functionality of an application, we will need to incorporate the entering of the username and the password. However, passwords are generally sensitive data that must be protected. Using UFT's Password Encoder tool, a test engineer is able to encrypt the password. Then, using the SetSecure method, the test engineer is able to enter the decrypted password into the password field in the application.

**Instead of having the following line of code:**
Browser("Browser1").Page("Page1").WebEdit("Password").Set "Password1234"

**We will have the following line of code:**
Browser("Browser1").Page("Page1").WebEdit("Password").SetSecure "55a93575b17558ee6540ee42e898de43aca0e76a"

The text being set is an encrypted text. The test engineer will not see the encrypted password, but UFT will be able to enter the encrypted password after decrypting it via the 'SetSecure' method.