Experiment 9: Apply the knowledge of test cases for the project using white box testing.

Learning Objective: Students will able to create unit test cases

Tools: Selenium IDe

Theory:

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

Unit Testing:

Unit testing focuses on the building blocks of the software system, that is, objects and subsystems. The specific candidates for unit testing are chosen from the object model and the system decomposition. In principle, all the objects developed during the development process should be tested, which is often not feasible because of time and budget constraints. The minimal set of objects to be tested should be the participating objects in the use cases. Subsystems should be tested after each of the objects and classes within that subsystem have been tested individually Unit testing focuses verification effort on the smallest unit of software design—the software component or module.The unit test is white-box oriented. . In Unit testing the following are tested,

1. The module interface is tested to ensure that information properly flows into and out of the program unit under test.
2. The local data structure is examined to ensure that data stored temporarily maintains its integrity.
3. Boundary conditions are tested to ensure that the module operates properly at boundaries established to limit or restrict processing.
4. All independent paths through the control structure are exercised to ensure that all statements in a module have been executed at least once.
5. And finally, all error handling paths are tested

Learning Outcomes: Students should have the ability to

LO1: Students will be able to understand Software Testing Concepts and the various Software standards.
L02:to test a software with the help of selenium id
LO3: create test cases

LO4: To understand different tools for testing

Outcomes: Upon completion of the course students will be able to write test cases for the project.

Conclusion:

We successfully wrote 10 test cases for the project. We successfully tested the website in selenium.

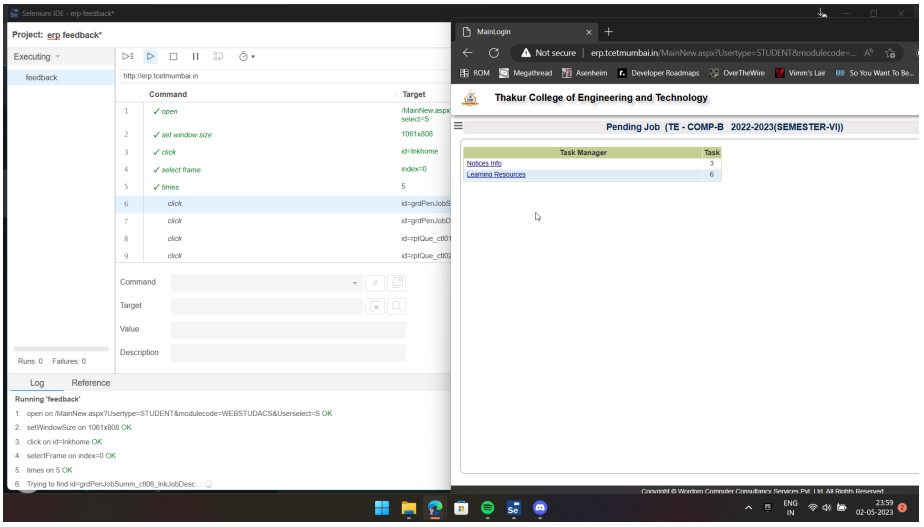
Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				

Code: # Generated by Selenium IDE

```
import pytest
import time
import json
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
from selenium.webdriver.support import expected_conditions
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.desired_capabilities import DesiredCapabilities

class TestFeedback():
  def setup_method(self, method):
    self.driver = webdriver.Chrome()
    self.vars = {}

  def teardown_method(self, method):
    self.driver.quit()
    def test_feedback(self):
```




```
self.driver.get("http://erp.tcetmumbai.in/MainNew.aspx?Usertype=STUDENT&modulecode=WEBSTUDACS&Userselect=S")
    self.driver.set_window_size(1061, 808)
    self.driver.find_element(By.ID, "lnkhome").click()
    self.driver.switch_to.frame(0)
    for i in range(0, 5):
      self.driver.find_element(By.ID, "grdPenJobSumm_ctl06_lnkJobDesc").click()
      self.driver.find_element(By.ID, "rptQue_ctl01_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl02_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl03_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl04_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl05_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl06_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl07_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl08_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl09_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl10_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl11_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "rptQue_ctl12_rptOpt_ctl00_chkSele").click()
      self.driver.find_element(By.ID, "btnSave").click()
      element = self.driver.find_element(By.ID, "btnSave")
      actions = ActionChains(self.driver)
      actions.move_to_element(element).perform()
      assert self.driver.switch_to.alert.text == "Are you sure to mark the feedback as over ?"
      self.driver.switch_to.alert.accept()
      self.driver.find_element(By.ID, "jsAlert1_okButton").click()
```