**Set 2**

**Question 1:**

public class Solution1 extends TestBase{

@BeforeMethod

@Parameters({ "browser" })

public void beforeMethod(String browser) throws Throwable {

init(browser, "https://exammay2020.agiletestingalliance.org/");

checkPageReady();

}

@Test

public void f() throws Throwable {

d.findElement(By.xpath("//\*[@class='eicon-close']")).click();

//checkPageReady();

mouseover(d, d.findElement(By.xpath("//\*[@id='menu-main-1']//\*[contains(text(),'Doppa2020')]")));

takeScreenShot(d, "Dotta");

WebElement e=d.findElement(By.xpath("//\*[@id='menu-main-1']//a[contains(text(),'About')]"));

jsClick(e);

wait(5);

List <WebElement> Names=d.findElements(By.xpath("//div[@class='elementor-row']//h4//parent::div"));

for(WebElement e1 : Names) {

//System.out.println();

String[] ar=e1.getText().split("\\r?\\n");

if(ar[1].toLowerCase().contains("Test".toLowerCase())) {

System.out.println(ar[0]);

}

}

scrollIntoView(d.findElement(By.className("social\_icon")));

takeScreenShot(d, "Social\_icon");

List<WebElement> socialIconLinks=d.findElements(By.xpath("//\*[@class='social\_icon']//a"));

String mainwindow=d.getWindowHandle();

for(WebElement icon : socialIconLinks) {

System.out.println(icon.getAttribute("href"));

icon.click();

Set <String> childwindow=d.getWindowHandles();

Iterator <String> winItr=childwindow.iterator();

while(winItr.hasNext()) {

String child=winItr.next();

if(!child.equals(mainwindow)) {

d.switchTo().window(child);

checkPageReady();

System.out.println(d.getTitle());

takeScreenShot(d, "Social\_Pages");

d.close();

}

d.switchTo().window(mainwindow);

}

}

List<WebElement> doppatext=d.findElements(By.xpath("//\*[contains(text(),'doppa2020')]"));

System.out.println("Doppa2020 is displayed : "+ doppatext.size());

}

**Question 2:**

public class Solution2 extends TestBase {

int counter=1;

@BeforeMethod

@Parameters({ "browser" })

public void beforeMethod(String browser) throws Throwable {

init(browser, "https://exammay2020.agiletestingalliance.org/");

waitForElementToVisible(d.findElement(By.xpath("//\*[@class='eicon-close']")));

d.findElement(By.xpath("//\*[@class='eicon-close']")).click();

mouseover(d, d.findElement(By.xpath("//ul[@id='menu-main-1']//a[text()='Data']")));

d.findElement(By.xpath("//ul[@id='menu-main-1']/li[4]/ul/li[2]/a[1]")).click();

checkPageReady();

}

@Test(dataProvider = "datapro")

public void f(String id, String EmployeeName, String city, String HireDate) throws Exception {

HashMap<String,String> act= new HashMap<String, String>();

act.put("ID", id);

act.put("EMPLOYEENAME", EmployeeName);

act.put("CITY", city);

act.put("HIREDATE", HireDate);

boolean flag = false;

boolean resultFlag=false;

int count=1;

int pageSize=d.findElements(By.xpath("//\*[@class='ea-advanced-data-table-pagination ea-advanced-data-table-pagination-button clearfix']//a")).size();

int pagecount=3;

int flag1=0;

int j=1;

while(j<=5) {

count=j;

String getID=d.findElement(By.xpath("//tbody//tr[@style='display: table-row;']["+j+"]//td[1]")).getText();

if (count==5) {

d.findElement(By.xpath("//\*[@class='ea-advanced-data-table-pagination ea-advanced-data-table-pagination-button clearfix']//a["+pagecount+"]")).click();

checkPageReady();

flag1=1;

j=0;

if(pageSize==pagecount +1) {

break;

}

pagecount++;

}

if(getID.equalsIgnoreCase(id.trim())){

resultFlag=true;

break;

}

j++;

}

String getID=null;

String getEmployeeName=null;

String getCity=null;

String getHireDate=null;

if(resultFlag) {

System.out.println("PASS");

getID=d.findElement(By.xpath("//tbody//tr[@style='display: table-row;']["+count+"]//td[1]")).getText();

getEmployeeName=d.findElement(By.xpath("//tbody//tr[@style='display: table-row;']["+count+"]//td[2]")).getText();

getCity=d.findElement(By.xpath("//tbody//tr[@style='display: table-row;']["+count+"]//td[3]")).getText();

getHireDate=d.findElement(By.xpath("//tbody//tr[@style='display: table-row;']["+count+"]//td[4]")).getText();

HashMap<String,String> exp=new HashMap<String,String>();

exp.put("ID", getID);

exp.put("EMPLOYEENAME", getEmployeeName);

exp.put("CITY", getCity);

exp.put("HIREDATE", getHireDate);

flag=exp.equals(act);

}

String result="";

if(flag) {

result="PASS";

}

else

{

result="FAIL";

}

setData(Constants.pathForWritingData, "Sheet1", counter, 0, id);

setData(Constants.pathForWritingData, "Sheet1", counter, 1, EmployeeName);

setData(Constants.pathForWritingData, "Sheet1", counter, 2, city);

setData(Constants.pathForWritingData, "Sheet1", counter, 3, HireDate);

setData(Constants.pathForWritingData, "Sheet1", counter, 4, result);

counter++;

}

@DataProvider

public Object[][] datapro() throws Exception {

Object[][] data = readData(Constants.TestDataExcel\_PATH, "Sheet1");

return data;

}

**Question 3:**

public class Solution3 extends TestBase{

@Before

public void beforeMethod() throws Throwable {

init("Chrome", "https://exammay2020.agiletestingalliance.org/data2");

checkPageReady();

}

@Test

public void f() throws InterruptedException {

Solution3pageObject obj=new Solution3pageObject(d);

obj.getDataSetName();

String[] Searchname = {"Vaishali", "Fabian", "World"};

for(int i = 0; i< Searchname.length; i++){

List<String> cha = obj.searchData(Searchname[i]);

System.out.println("Search Key: "+Searchname[i]+" data: "+cha);

}

}

**POM**

public class Solution3pageObject extends TestBase {

WebDriver d;

public Solution3pageObject(WebDriver d) {

this.d = d;

}

By header=By.xpath("//h2[contains(text(),'DATA SET 2')]");

By searchBox=By.xpath("//input[@class='ea-advanced-data-table-search']");

By searchResult=By.xpath("//tbody//tr[@style='display: table-row;']");

public void getDataSetName() {

String heading=d.findElement(header).getText();

System.out.println("Header is: "+heading);

}

public List<String> searchData(String keyword) {

List<String> arr = new ArrayList<String>();

d.findElement(searchBox).clear();

d.findElement(searchBox).sendKeys(keyword);

List <WebElement> results=d.findElements(searchResult);

//System.out.println("Records found= " +results.size());

if (results.size()>0) {

for(WebElement e: results) {

//System.out.println(e.getText());

arr.add(e.getText());

}

}else {

arr.add("No Data Found");

}

return arr;

}

}

**Question 4:**

public class Solutions4 extends TestBase {

int counter = 1;

@BeforeMethod

@Parameters({ "browser" })

public void launchApplication() throws Throwable {

init("Chrome", "https://exammay2020.agiletestingalliance.org/");

waitForElementToVisible(d.findElement(By.className("eicon-close")));

d.findElement(By.className("eicon-close")).click();

// mouseover(d, d.findElement(By.xpath("(//ul[@id='menu-main-1']//a[text()='Pages'])")));

checkPageReady();

}

@Test()

public void appTest() throws Throwable {

mouseover(d, d.findElement(By.xpath("(//ul[@id='menu-main-1']//a[text()='Pages'])")));

d.findElement(By.xpath("(//\*[@id='menu-main-1']/li[5]/ul/li[2]/a)[1]")).click();

WebElement tickersSection = d.findElement(By.xpath("//ul[@class='eael-tab-inline-icon']"));

scrollIntoView(tickersSection);

d.findElement(By.className("eicon-close")).click();

takeScreenShot(d, "Question4\_Upcoming Meetups and Events");

List<WebElement> tickers = d.findElements(By.xpath("//div[@class='ticker-content']//a"));

String tickerMsg = "";

boolean flag = false;

for (WebElement ticker : tickers) {

String theTextIWant = ticker.getAttribute("textContent");

tickerMsg = theTextIWant + " ,";

System.out.println(theTextIWant);

if (tickerMsg.contains(theTextIWant)) {

flag = true;

}

}

if (!flag) {

System.out.println("Messages are not matching");

}

for (WebElement ticker : tickers) {

if (ticker.getAttribute("href") == null || ticker.getAttribute("href") == "") {

} else {

System.out.println(ticker.getAttribute("href"));

}

}

int count = 1;

List<WebElement> plusIcon = d

.findElements(By.xpath("//i[@class='fa-toggle fas fa-angle-down']//preceding-sibling::i"));

for (WebElement ele : plusIcon) {

String beforeClick = ele.getCssValue("color");

String beforeClick1 = Color.fromString(beforeClick).asRgb();

jsClick(ele);

String afterClick = ele.getCssValue("color");

String afterClick1 = Color.fromString(afterClick).asRgb();

wait(2);

System.out.println(

d.findElement(By.xpath("(//div[contains(@id,'elementor-tab-conten')]//p[2])[" + count + "]"))

.getText());

System.out.println();

count++;

assertNotSame(beforeClick1, afterClick1, "not same");

System.out.println("Before Click RGB Color : " + beforeClick1);

System.out.println("After Click RGB Color : " + afterClick1);

}

}