

一、基础题

选择题

1.[多选]针对标签定位下面说法正确的是:

```
<body>
  <input type="text" class="inp_fmobil J_cellphone" name="invite">
  <input class="inp_fsecpass J_password2" id="password2" name="password2" >
</body>
```

- ☒ A.使用driver.find_element_by_tag_name("input")可以定位第一个input元素;
- ☒ B.使用driver.find_element_by_xpath("//input")可以定位第一个input元素;
- ☒ C.不建议使用标签定位, 因为一般情况下每个标签在页面中存在多个的可能性较大;
- ☒ D.页面上所有的元素都有标签名。

2.[多选]对于下面的元素, 能成功定位的表达式有哪些?

```
<body>
  <div class="item-left">
    <h3 class="cata-nav-name">
      <div class="cata-nav-wrap">
        <i class="ico ico-nav-1"></i>
        <a href="/Home/Goods/goodsList/id/2.html" title="家用电器">家用电器
      </a>
    </div>
  </h3>
</div>
</body>
```

- ☒ A.driver.find_element_by_link_text("家用电器");
- ☒ B.driver.find_element_by_partial_link_text("电器")
- ☒ C.driver.find_element_by_tag_name("a")
- ☒ D.driver.find_element_by_xpath("//a")
- ☒ E.driver.find_element_by_xpath("//div/a")
- ☒ F.driver.find_element_by_xpath("//*[@title='家用电器']")
- ☐ G.driver.find_element_by_xpath("//*[text()='家用']")

3.[多选]:对于xpath下面说法正确的是?

- ☒ A.xpath是一种标记语言, 可以用于在html中进行元素查找;
- ☒ B.xpath可以使用标签内的任意元素来进行定位;
- ☒ C.xpath可以基本解决所有元素定位的问题;
- ☒ D.xpath定位方式将整个页面的所有元素进行扫描以定位我们所需要的元素, 如果脚本中大量使用xpath做元素定位的话, 脚本的执行速度可能会稍慢。

4.[多选]:定位下面 input的元素, xpath表达式正确是?

```
<body>
  <div id='testA'>
    <input type="text" class="inp fmobile J_cellphone" name="invite"
id="password">
  </div>
</body>
```

- ☐ A.//*[@input="input"]
- ☒ B.//input[@class='inp fmobile J_cellphone']
- ☐ C./div/input
- ☒ D.//*[@id='password']

5.[多选]:定位标签待付款, 对于xpath属性定位表达式正确的是:

```
<div class="navitems2 p" id="navitems5">
  <ul>
    <li>
      <a href="order_list.html" value="全部订单" class="selected">全部订单
</a>
    </li>
    <li>
      <a href="WAITPAY.html" value="待付款" class="">待付款</a>
    </li>
    <li>
      <a href="WAITSEND.html" value="待发货" class="">待发货</a>
    </li>
  </ul>
</div>
```

- ☒ A.//a[@value='待付款']
- ☒ B.//*[@value='待付款']
- ☒ C.//a[@href='WAITPAY.html']
- ☒ D.//*[@value='待付款' and @href='WAITPAY.html']

6.[多选]:定位全部订单付款, 对于xpath属性层级表达式正确的是:

```
<div class="navitems2 p" id="navitems5">
  <ul class="tb1">
    <li id="table1">
      <a href="order_list.html" value="全部订单" class="selected">全部订单
</a>
    </li>
  </ul>
</div>
```

- ☒ A. //ul/li/a
- ☒ B.//li/a
- ☒ C.//li[@id='table1']/a
- ☐ D.//ul[@class='tb1']/a[1]

7.[多选]:关于CSS说法正确的是:

- ☒ CSS定位相对xpath来讲的速度更快;
- ☒ CSS定位时通过CSS选择器的模式来实现选择元素的;
- ☒ CSS是用来描述HTML样式语言;
- ☐ CSS不能定位一组元素;

8.[多选]:以下CSS表达式错误的是:

```
<div class="navitems2 p" id="navitems5">
  <input id="testA" name="testA" value="all" class="selected">请输入</input>
</div>
```

- ☐ A.#testA
- ☒ B.*testA
- ☐ C. .selected
- ☒ D.[@value='all']
- ☐ E. input
- ☐ F.input[value='all']

9.[多选]:通过CSS层级选择器能定位到input标签元素的表达式有哪些:

```
<div class="p" id="navitems5">
  <div>
    <div id="testA" name="testA" value="all"/>
  </div>
  <div id="demoA">
    <input id="testA" name="testA" value="all" class="selected">请输入
  </input>
  </div>
</div>
```

- ☐ A.div [@id='testA']
- ☒ B.div>#testA
- ☒ C.div[class='p'] input
- ☒ D.#demoA .selected

10.下面关于元素定位说法错误是:

- ☐ A.driver.find_element_by_tag_name("input")和driver.find_element_by_xpath("//input")定位结果一样;
- ☐ B.driver.find_element_by_xpath("//input")和driver.find_element_by_css_selector("input")定位结果一样;
- ☐ C.CSS层级定位策略中层级选择器, 空格表示法可以代替">"的表示法;
- ☒ D.CSS可以定位到所有的元素;

11.[多选]定位下面html中所有input标签, 使用方法正确的是:

```

<div>
  <div>
    <input id="username" name="username" value="all" class="usrA">请输入用户名
  </div>
  <div id="demoA">
    <input id="password" name="password" value="all" class="pwdA">请输入密码
  </div>
  <div id="demoA">
    <input id="phone" name="phone" value="all" class="phone">请输入电话号码
  </div>
</div>

```

- ☒ A.driver.find_elements(By.CSS_SELECTOR,"input")
- ☒ B.driver.find_elements_by_tag_name("input")
- ☐ C.driver.find_elements_by_input("input")
- ☒ D.driver.find_elements_by_xpath("//input")

12.[多选]请选择定位下面的input元素的方式效果一样的选项:

```

<div class="navitems2 p" id="navitems5">
  <input id="testA" name="testA" value="all" class="selected">请输入</input>
</div>

```

- ☒ A.driver.find_element_by_id("testA") 和 driver.find_element_by_class_name("selected")
- ☒ B.driver.find_element(By.XPATH,"//input") 和 driver.find_element_by_css_selector("div input")
- ☐ C.driver.find_elements(By.CSS_SELECTOR,"#testA") 和 driver.find_element_by_name("testA")
- ☒ D.driver.find_element_by_xpath("//div[@class='p']/input") 和 driver.find_element_by_xpath("//input")

13.[多选]选择以下关于浏览器驱动对象常用方法说明正确的选项:

- ☒ A.浏览器最大化窗口driver.maximize_window()必须写在实例化浏览器驱动对象之后;
- ☒ B.可以通过浏览器驱动对象的属性driver.title来获取当前页面标题;
- ☒ C.可以通过浏览器驱动对象的属性driver.current_url来获取当前页面网址;
- ☒ D.所有的元素定位的方法都是浏览器驱动对象所提供的。

14.[多选]以下关于driver.close()和driver.quit()区别说法正确的是:

- ☒ A.driver.quit()会关闭所相关的窗口, 退出驱动;
- ☒ B.driver.close()关闭当前窗口, 如果当前打开的是最后一个窗口, 则退出浏览器;
- ☒ C.driver.close()当前打开的是最后一个窗口, 则退出浏览器, 但是浏览器驱动进程不会关闭;
- ☐ D.同一个浏览器驱动对象下面可以存在多个当前页面;

代码题

1.按以下要求完成登陆流程:

```
# 1. 导包
```

```

from selenium import webdriver
# 2.实例化浏览器对象
driver = webdriver.Chrome()
# 3.打开测试网址
driver.get('http://localhost/Home/user/reg.html')
# 4.业务操作
# 1.使用css定位定位tpshop首页登录超链接，并执行点击
driver.find_element_by_css_selector(".red").click()
# 2.使用id定位定位登陆页面用户名输入框，输入用户名
driver.find_element_by_id("username").send_keys("15800000001")
# 3.使用name定位定位登陆页面密码输入框，输入密码
driver.find_element_by_name("password").send_keys("123456")
# 4.使用class定位定位登陆页面验证码输入框，输入验证码
driver.find_element_by_class_name("text_cmu").send_keys("8888")
# 5.使用xpath定位登陆按钮执行点击
driver.find_element_by_xpath("//a[@name='sbtbutton']")
# 6.获取当前页面标题和地址信息并打印
title = driver.title
url = driver.current_url
print(title,url)
# 5.关闭浏览器
driver.quit()

```

2.按以下要求完成注册流程:

```

# 1.导包
from selenium import webdriver
from selenium.webdriver.common.by import By

# 2.实例化浏览器对象
driver = webdriver.Chrome()
# 3.打开测试网址
driver.get('http://localhost/')
# 4.业务操作
# 要求使用元素定位另外一种写法: driver.find_element(By.XXX,value)
# 1.使用xpath的文本形式定位tpshop首页注册超链接，并点击
driver.find_element(By.XPATH, "//*[text()='注册']").click()
# 2.使用css属性定位策略定位手机号码输入框，输入手机号码
driver.find_element(By.CSS_SELECTOR, "[name='username']").send_keys('15800000005')
# 3.使用css属性包含的方式定位验证码输入框，输入验证码
driver.find_element(By.CSS_SELECTOR, "[placeholder*='图像']").send_keys('123456')
# 4.使用xpath属性与逻辑结合定位方式定位密码输入框，输入设置密码
driver.find_element(By.XPATH, '//*[type="password" and @name="password"]').send_keys('123456')
# 5.使用xpath路径定位策略定位确认密码，输入确认密码
driver.find_element(By.XPATH, "//div/div/input[@name='password2']").send_keys('123456')
# 6.使用超链接定位方式定位同意协议并注册按钮，并点击
driver.find_element(By.LINK_TEXT, "同意协议并注册").click()
# 5.关闭浏览器
driver.quit()

```

二、提高题

1.针对标签定位下面说法正确的是:

```
<body>
  <input type="text" class="inp_fmobil e_j_cellphone" name="invite">测试</input>
</body>
```

- ☐ A.如body中只有以上input标签包含name属性则通过CSS表达式 "[invite]" 可以定位到该元素;
- ☐ B.xpath表达式: `//*[contains(@class,"cellphone")]` 和 CSS表达式: `[class='cellphone']`表示的意识一样;
- ☐ C.可以通过xpath: `//*[contains(text(),"测试")]`找定位到input标签元素;
- ☒ D.以上都对。

2.如下html, class属性每次打开页面后面数字都会发生改变, 如需要定位定位第三个input标签, 下面表达式正确的是:

```
<!--class标签每次都打开都会发生改变-->
<body>
  <input type="text" class="test001" name="invite">测试</input>
  <input type="text" class="test002" name="invite">测试</input>
  <input type="text" class="test003" name="invite">测试</input>
  <input type="text" class="test004" name="invite">测试</input>
</body>
```

- ☒ A.driver.find_elements(By.XPATH,"//*[contains(@class,'test')]")[2]
- ☒ B.driver.find_elements_by_tag_name("input")[2]
- ☒ C.driver.find_element(By.XPATH,"//input[3]")
- ☒ D.driver.find_elements_by_css_selector("[name='invite']")[2]
- ☒ E.driver.find_elements_by_xpath("//*[text()='测试']")[2]

3.请选择下面说法正确的选项;

- ☐ A.打开浏览器测试页面后如果未最大化可能出现窗口边缘未渲染导致元素定位不到;
- ☐ B.driver.refresh()后会重新发送请求到web后台服务器;
- ☐ C.当前浏览器只打开的一个窗口执行driver.close()再执行driver.quit()不会报错, 但是先执行driver.quit()再执行driver.close()会报错;
- ☒ D.以上说法都正确;