# 选择框操作的基本方法

在 Selenium 自动化测试中，操作选择框（如下拉菜单、单选框、复选框）是常见需求。以下是选择框操作的基本方法及示例（以 Python 为例）：

### 一、下拉菜单（Select 元素）

对于标准的 <select> 标签，使用 Selenium 的 Select 类操作。

#### 1. 导入 Select 类

from selenium.webdriver.support.ui import Select

#### 2. 初始化 Select 对象

select\_element = driver.find\_element(By.ID, 'country')  
select = Select(select\_element)

#### 3. 选择选项

# 通过 value 属性选择  
select.select\_by\_value('us')  
  
# 通过可见文本选择  
select.select\_by\_visible\_text('美国')  
  
# 通过索引选择（从 0 开始）  
select.select\_by\_index(2)

#### 4. 获取当前选中项

# 获取第一个选中项的文本  
selected\_option = select.first\_selected\_option.text  
print(selected\_option) # 输出：美国  
  
# 获取所有选中项（支持多选）  
all\_selected\_options = [option.text for option in select.all\_selected\_options]

#### 5. 获取所有选项

all\_options = [option.text for option in select.options]  
print(all\_options) # 输出所有选项文本

#### 6. 多选框操作

# 先取消所有选择  
select.deselect\_all()  
  
# 选择多个选项  
select.select\_by\_value('option1')  
select.select\_by\_value('option2')  
  
# 取消特定选项  
select.deselect\_by\_value('option1')

### 二、自定义下拉菜单

对于非标准的 <select> 元素（如 <div> 模拟的下拉菜单），需手动点击触发并选择。

#### 示例：操作自定义下拉菜单

# 点击下拉菜单触发按钮  
dropdown\_button = driver.find\_element(By.ID, 'dropdown-trigger')  
dropdown\_button.click()  
  
# 等待选项加载  
from selenium.webdriver.support.ui import WebDriverWait  
from selenium.webdriver.support import expected\_conditions as EC  
  
options = WebDriverWait(driver, 10).until(  
 EC.presence\_of\_all\_elements\_located((By.CSS\_SELECTOR, '.dropdown-option'))  
)  
  
# 选择特定选项  
for option in options:  
 if option.text == '目标选项':  
 option.click()  
 break

### 三、单选框（Radio Button）

单选框通过 click() 方法选择，同一组单选框只能选中一个。

#### 示例：选择单选框

# 选择性别为 "男"  
male\_radio = driver.find\_element(By.CSS\_SELECTOR, 'input[name="gender"][value="male"]')  
male\_radio.click()  
  
# 检查是否已选中  
if male\_radio.is\_selected():  
 print("已选中男性")

### 四、复选框（Checkbox）

复选框通过 click() 方法切换选中状态。

#### 示例：操作复选框

# 选择 "我同意条款"  
agree\_checkbox = driver.find\_element(By.ID, 'agree')  
  
# 如果未选中，则点击选中  
if not agree\_checkbox.is\_selected():  
 agree\_checkbox.click()  
  
# 取消选中  
agree\_checkbox.click()

### 五、动态加载的选择框

对于需要等待加载的选择框，使用显式等待。

#### 示例：等待下拉菜单选项加载

from selenium.webdriver.support.ui import WebDriverWait  
from selenium.webdriver.support import expected\_conditions as EC  
  
# 点击下拉菜单  
driver.find\_element(By.ID, 'dynamic-dropdown').click()  
  
# 等待选项出现并选择  
option = WebDriverWait(driver, 10).until(  
 EC.element\_to\_be\_clickable((By.CSS\_SELECTOR, '.dropdown-item[data-value="target"]'))  
)  
option.click()

### 注意事项

1. **Select 类的限制**：

* 1. 仅适用于标准的 <select> 标签。

* 1. 对于自定义下拉菜单，需手动模拟用户操作。

1. **选项值与文本的区别**：

* 1. select\_by\_value() 使用 HTML 的 value 属性。

* 1. select\_by\_visible\_text() 使用用户可见的文本。

1. **等待机制**：

* 1. 动态加载的选择框需等待选项可见或可点击。

* 1. 使用 WebDriverWait 避免因加载延迟导致的元素找不到异常。

掌握这些方法后，就能处理各种类型的选择框，实现表单填写、筛选条件设置等自动化操作。