

Task 2

1. More generic code can be written. In other words, functions can be defined to apply to arguments of different types.

- Scenario
To check if the type of two target variables are the same. And try two examples as follow.
- Code Segment

```
def check_type(target1, target2):  
    if type(target1) == type(target2):  
        return True  
    else:  
        return False
```

```
if check_type("you", "me"):  
    print("same")  
if not check_type(3.14, 3):  
    print("different")
```

- Result

```
same  
different
```

2. Possibilities of mixed type collection data structures.

- Scenario
Assume list is a 2d array and each row stores the name, phone number and phone balance. And just see what is the type of each item of a row.
- Code Segment

```
list = [[] for i in range(100)]
```

```
list[0].append("Name")  
list[0].append(12345678)  
list[0].append(100.00)  
print(list[0])  
for obj in list[0]:  
    print(type(obj))
```

- Result

```
['Name', 12345678, 100.0]
<class 'str'>
<class 'int'>
<class 'float'>
```

- Scenario

To check if the type of two target variables are the same.

- Code Segment

```
def check_type(target1, target2):
    if type(target1) == type(target2):
        return True
    else:
        return False
```

```
if check_type("you", "me"):
    print("same")
if not check_type(3.14, 3):
    print("different")
```

- Result

```
same
different
```