

## Activity 3 - Code Review Hands One

### Code Snippets to Review

You will need to provide feedback as demonstrated in the assignment description for the following five snippets. Please use this template document to complete this assignment. You will need to provide written feedback and also “corrected” code just like in the examples in the assignment description.

**Name:** Hyunjae Kim

**email:** kimh6@oregonstate.edu

---

#### Snippet 1

*# Pull request 1*

```
def my_func(x):
```

```
    return x**2
```

#### Feedback:

- Feedback:

The code is simple, and follows the style guide. However, it is hard to understand what this function is doing. The name of the function provides no clue, and the function seems to returning the square of ‘x’. If this is the case, then the variable would benefit from clearer names.

- Correction:

```
def square(x):
    square = x**2

    return square
```

---

#### Snippet 2

*# Pull request 2*

```
def create_odds(num):
```

```
    """Creates a list of len(num) of random odd numbers between 1 and 1000"""
```

```
    num_list = []
```

```
    for i in range(0, num):
```

```
        new_num = 2
```

```
        while new_num % 2 == 0:
```

```
            new_num = random.randint(1, 1000)
```

```
    num_list.append(new_num)
return num_list
```

**def** create\_evens(num):

*"""Creates a list of len(num) of random even numbers between 1 and 1000"""*

```
num_list = []
for i in range(0, num):
    new_num = 1
    while new_num % 2 != 0:
        new_num = random.randint(1, 1000)
    num_list.append(new_num)
return num_list
```

### Feedback:

- Feedback:

Your code has clear names, and clearly demonstrates what the functions are doing. Good job in finding that “range()” is not including “num”, and starting the “i” from 0. However, your code has so much unnecessary parts and using unnecessary computing power in order to generate random odd or even numbers. You can make your functions much simpler by using feature of odd or even numbers. See my suggested code below.

- Correction:

**def** create\_odds(num):

*"""Creates a list of len(num) of random odd numbers between 1 and 1000"""*

```
num_list = []
for i in range(0, num):
    new_num = random.randint(0, 499)

    new_num = 2*new_num + 1
    num_list.append(new_num)
return num_list
```

**def** create\_evens(num):

*"""Creates a list of len(num) of random even numbers between 1 and 1000"""*

```
num_list = []
for i in range(0, num):
    new_num = random.randint(1, 500)

    new_num = 2*new_num
    num_list.append(new_num)
return num_list
```

---

### Snippet 3

*# Pull request 3*

```
def check_for_val(self, val):  
    """This member function checks to see if val exists in the class member  
    values and returns True if found"""  
    for i in range(len(self.values)):  
        if self.values[i] == val:  
            return True  
    return False
```

#### Feedback:

- Feedback:

Your code has clear expressions on demonstrating what the function is doing. However, your code is not pythonic enough. Instead of manually comparing each index of array, it's a good idea to simplify your code and make your code pythonic by using "in" feature in python. See the suggested code below.

- Correction:

```
def check_for_val(self, val):  
    """This member function checks to see if val exists in the class member  
    values and returns True if found"""  
    if val in self.values:  
        return True  
    return False
```

---

### Snippet 4

*# Pull request 4*

```
def get_val_index(arr, val):  
    """Searches arr for val and returns the index if found, otherwise -1"""  
    index = -1  
    for i in range(len(arr)):  
        if arr[i] == val:  
            index = i  
            break  
    return index
```

#### Feedback:

- Feedback:

Good job in clear function name and clear comment explaining what the function is doing. Moreover, your method to get the index of the value in array is mostly correct. However, your method is not pythonic way. Instead of manually comparing each index of array, it's a good idea to simplify your code and make your code pythonic by using "in" and ".index()" feature in python. See the suggested code below.

- Correction:

```
def get_val_index(arr, val):  
    """Searches arr for val and returns the index if found, otherwise -1"""  
    index = -1  
    if val in arr:  
        index = arr.index(val)  
    return index
```

---

## Snippet 5

*# Pull request 5*

*# This is a global variable used by other functions, do not change*

```
int_arr = [1, 2, 5, 2, 10, 45, 9, 100]
```

```
def print_sorted(arr):  
    """Prints the items in the array after sorting"""  
    arr.sort()  
    for num in arr:  
        print(num)
```

## Feedback:

- Feedback:

The code follows the style guide. I can find that what the function is generally doing. However, the name of the function is ambiguous. There is no clue if the function is sorting the array in ascending order or descending order. If you meant to print the numbers in ascending order, then the name of function would benefit from clearer name. See my suggested code below.

- Correction:

```
int_arr = [1, 2, 5, 2, 10, 45, 9, 100]  
  
def print_in_ascending_order(arr):  
    """Prints the items in the array after sorting"""  
    arr.sort()  
    for num in arr:  
        print(num)
```