**Thumati Pavan Venkata Narendra Kumar**

**EMP-ID-289219**

**PYTHON-PROJECT-1**

**1.Python Functions.**

* 1. **Example Python Code for User-Defined function**

****

* 1. **Example Python Code for calling a function**

****

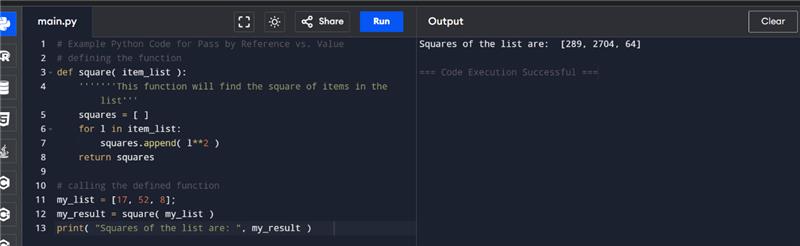
* 1. **Example Python Code for User-Defined function**

****

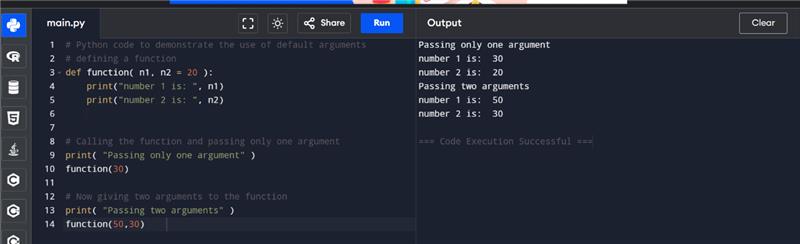
* 1. **Example Python Code for calling a function**

****

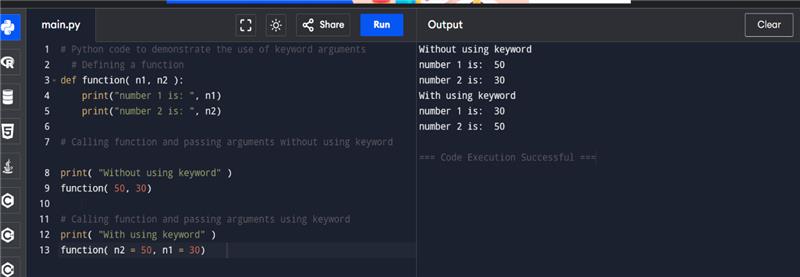
* 1. **Pass by Reference vs. Pass by Value**

****

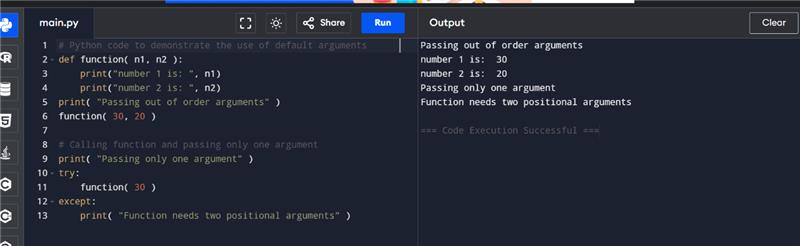
* 1. **Default Arguments**

****

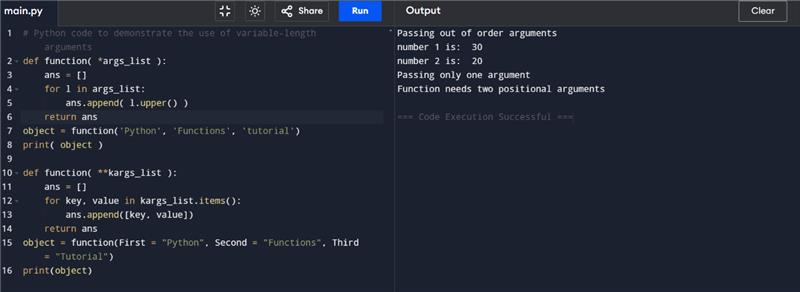
* 1. **Keyword Arguments**

****

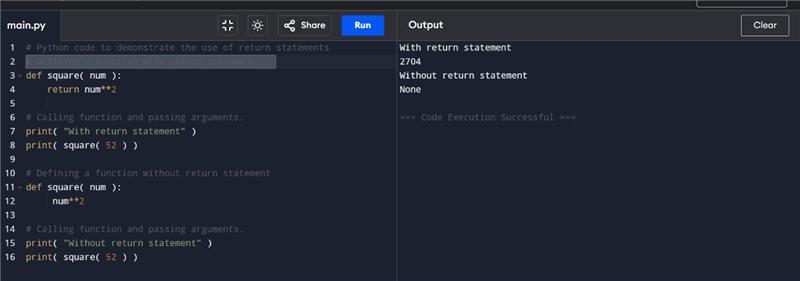
* 1. **Required Arguments**

****

* 1. **Variable-Length Arguments**

****

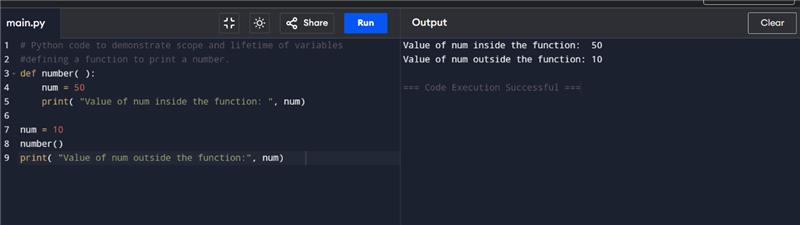
* 1. **Return statement**

****

* 1. **The Anonymous Functions**

****

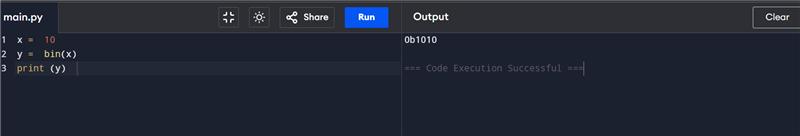
* 1. **Scope and Lifetime of Variables**

****

* 1. **Python abs() Function**

****

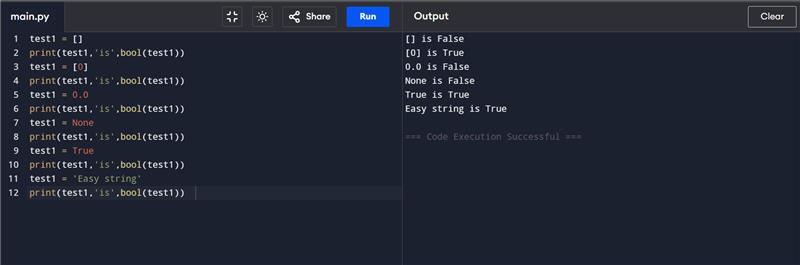
* 1. **Python bin() Function**

****

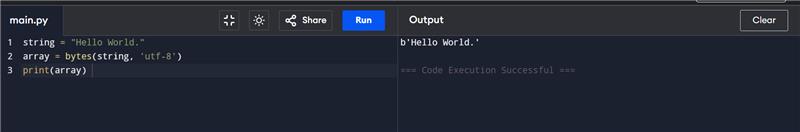
* 1. **Python all() Function**

****

* 1. **Python bool()**

****

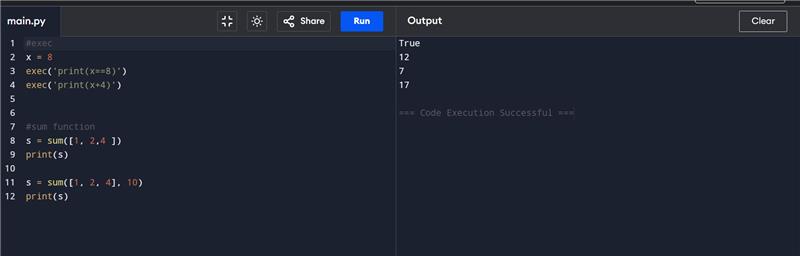
* 1. **Python bytes()**

****

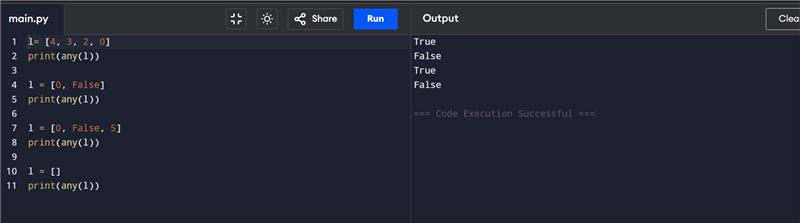
* 1. **Python compile() Function**

****

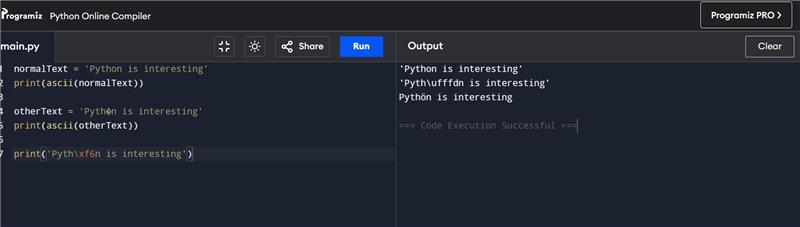
* 1. **Python exec() and sum() Function Example**

****

* 1. **Any() function**

****

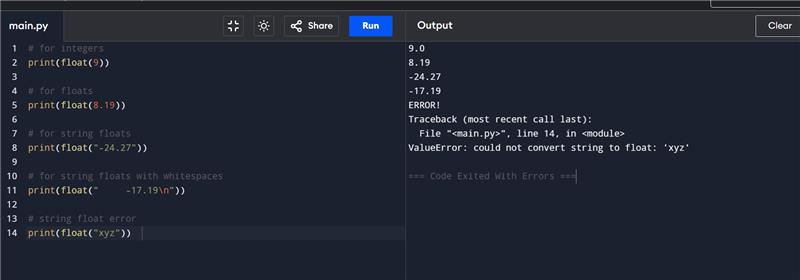
* 1. **ASCII() function**

****

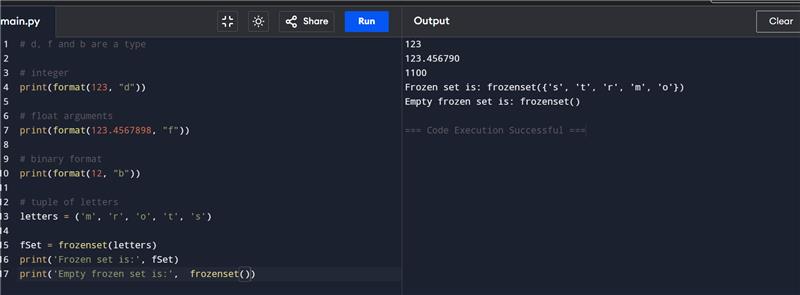
* 1. **Byte array() and eval() functions**

****

* 1. **Float() function**

****

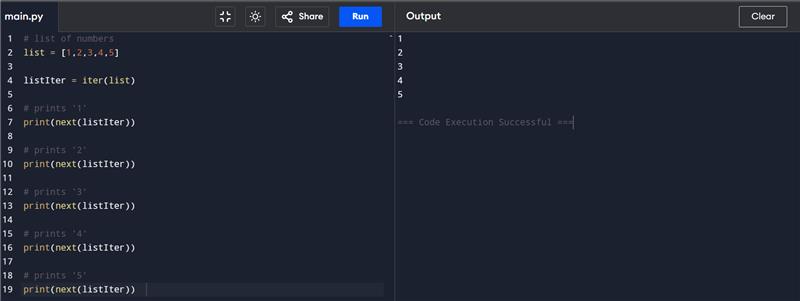
* 1. **Frozen set() and format() fucntion**

****

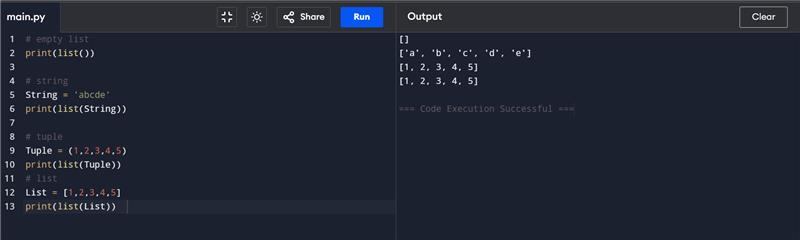
* 1. **Gloabls() and getattr() function**

****

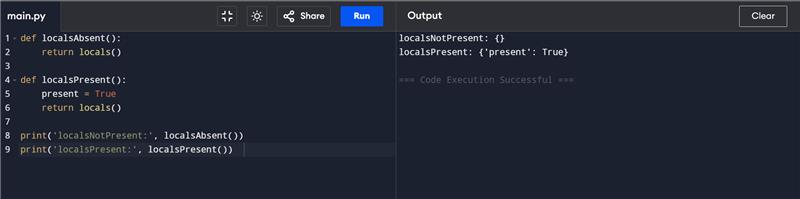
* 1. **Python iter() Function**

****

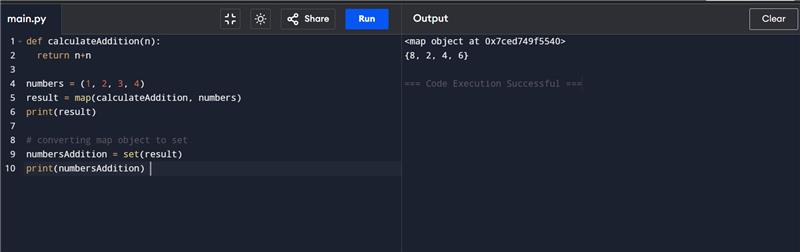
* 1. **Python list()**

****

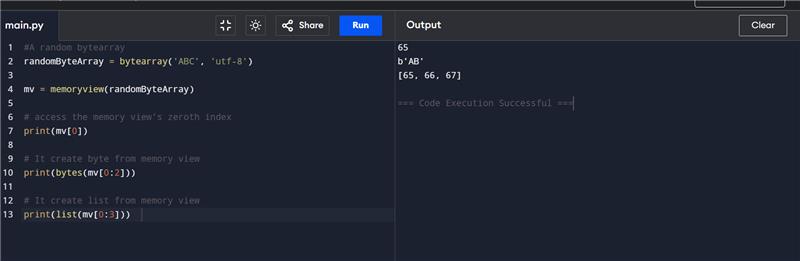
* 1. **Python locals() Function Example**

****

* 1. **Map() function**

****

* 1. **Python memoryview() Function**

****

* 1. **Python chr() Function**

****

* 1. **Python complex fun()**

****

* 1. **Python delattr() Function**

****

* 1. **Python enum()**

****

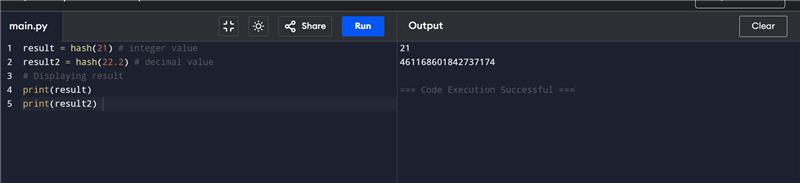
* 1. **Python dict()**

****

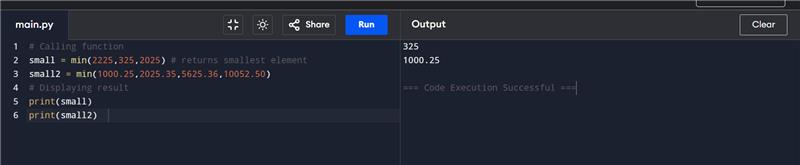
* 1. **Python filter ()**

****

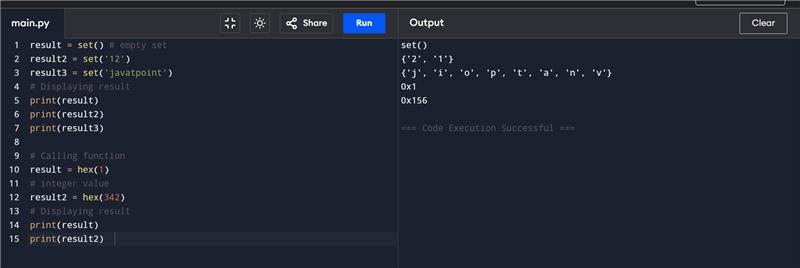
* 1. **Python hash()**

****

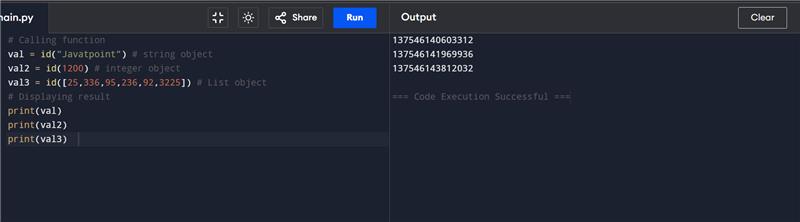
* 1. **Python min()**

****

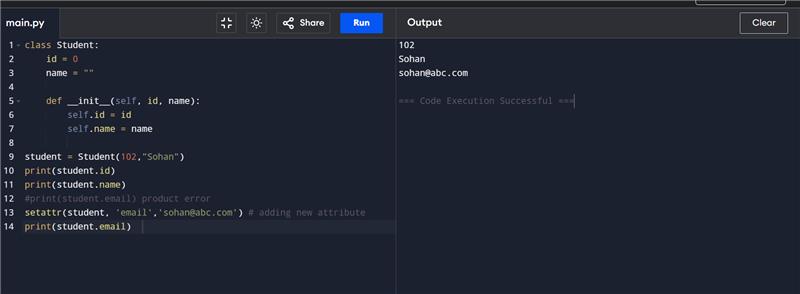
* 1. **Python hex() and set() function**

****

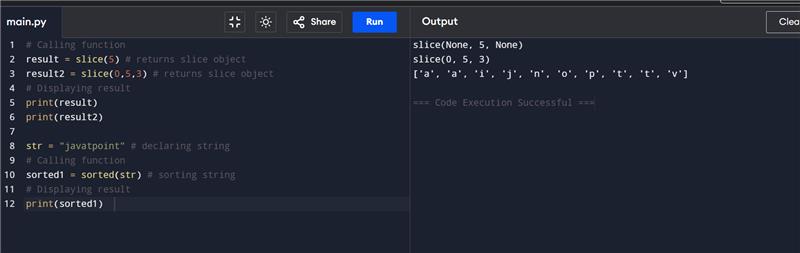
* 1. **Python Id()**

****

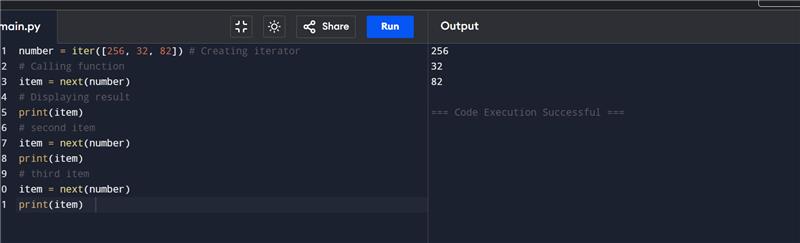
* 1. **Python setattr()**

****

* 1. **Python slice() and sorted()**

****

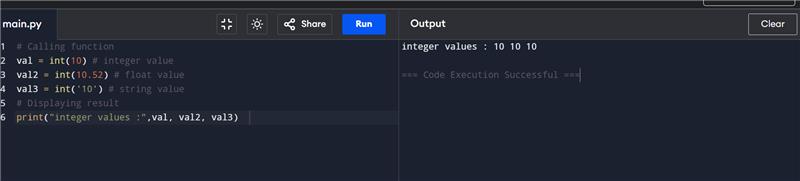
* 1. **Python next()**

****

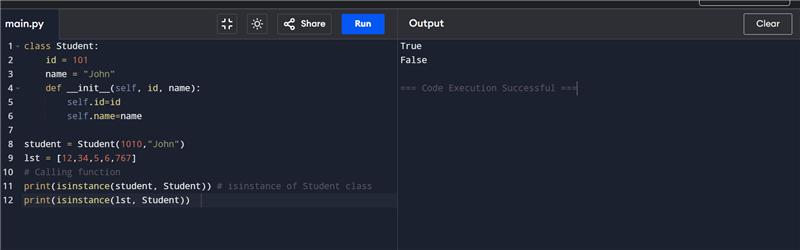
* 1. **Python input()**

****

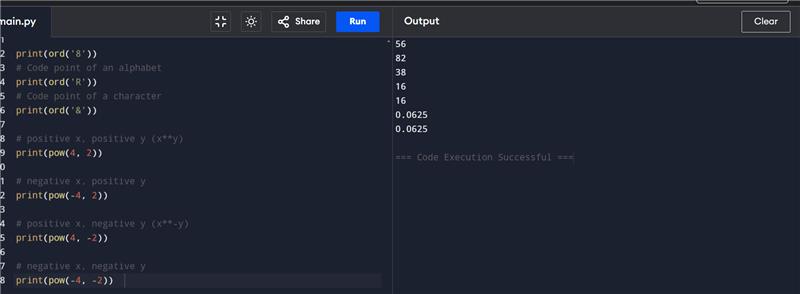
* 1. **Python int()**

****

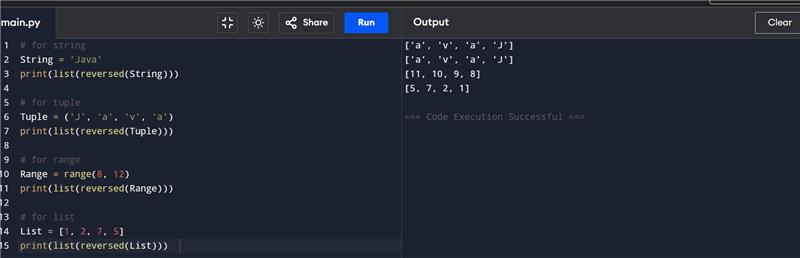
* 1. **Python instance()**

****

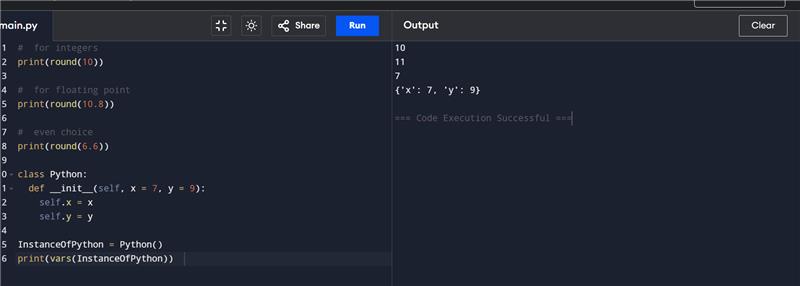
* 1. **Python ord() and pow() function**

****

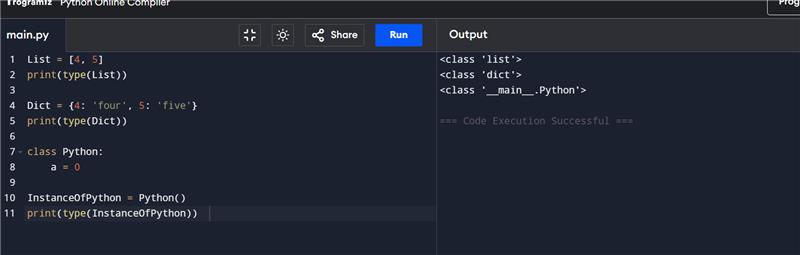
* 1. **Python reversed()**

****

* 1. **Python round() and var()**

****

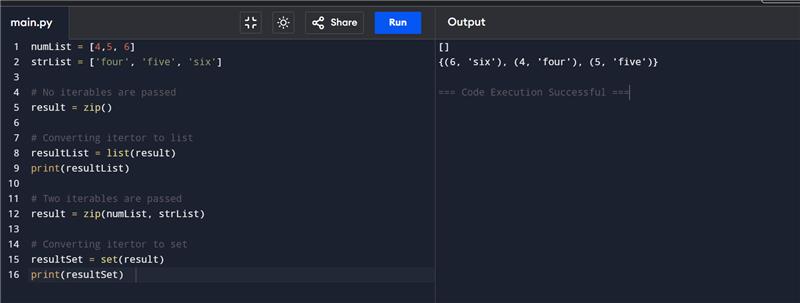
* 1. **Python type() fucntion**

****

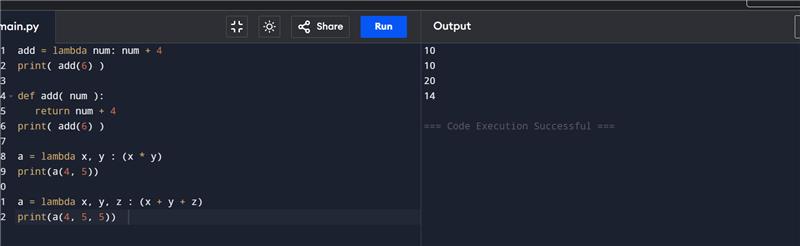
* 1. **Python issubclass()**

****

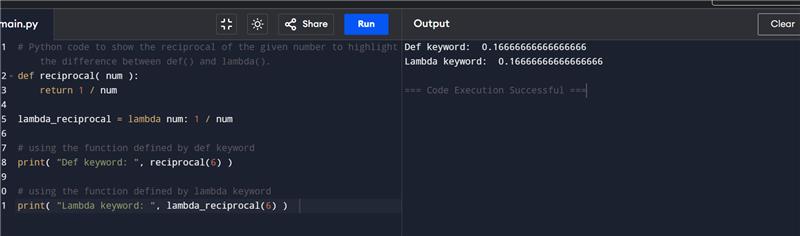
* 1. **Python zip()**

****

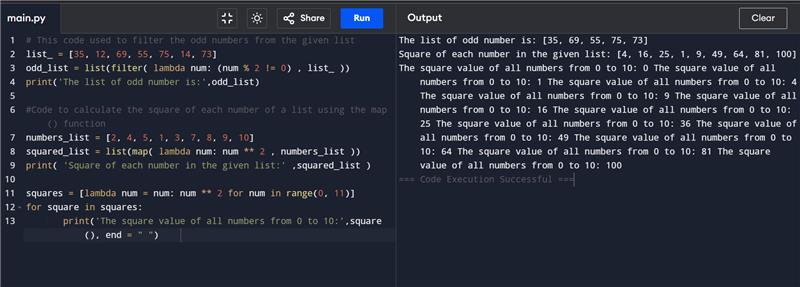
* 1. **Python lambda()**

****

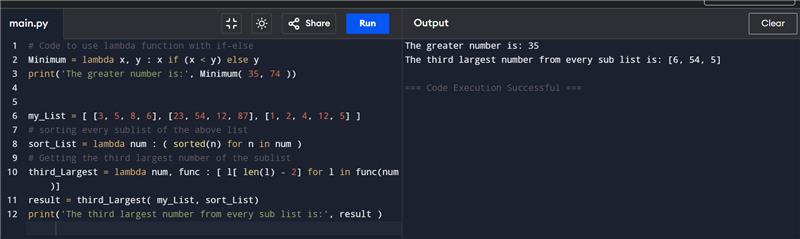
* 1. **Def vs lambda difference**

****

* 1. **Using Lambda Function with filter(),map(),list comprehension()**

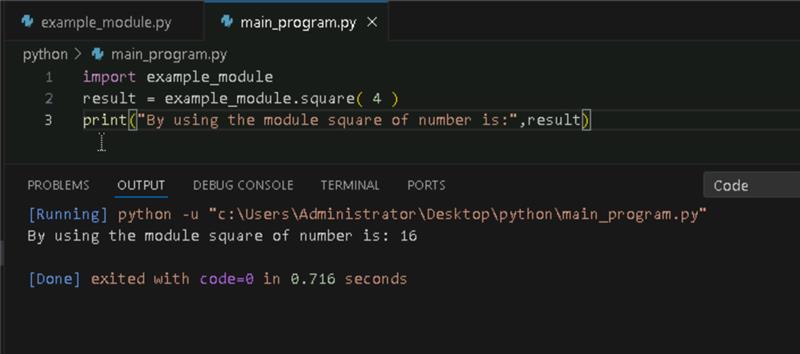
****

* 1. **Lambda function with multiple statement and if else**

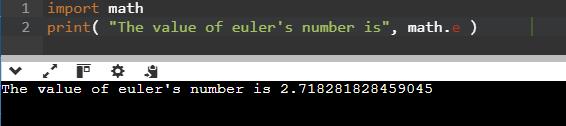
****

**2.MODULES**

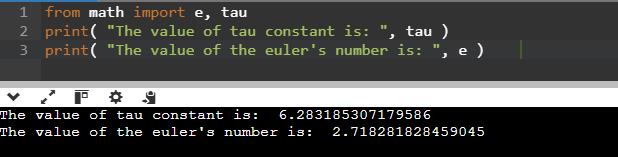
**2.1. Importing modules**

****

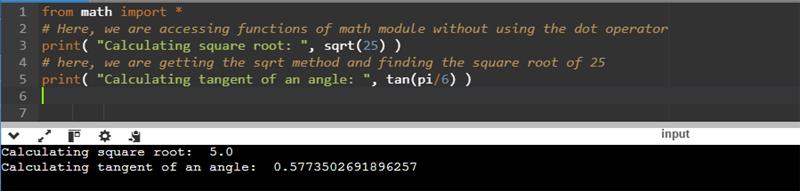
**2.2. Importing and also Renaming:**

****

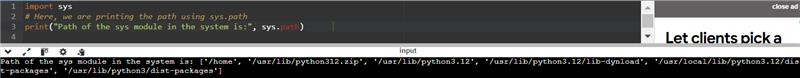
**2.3. Python from...import Statement:**

****

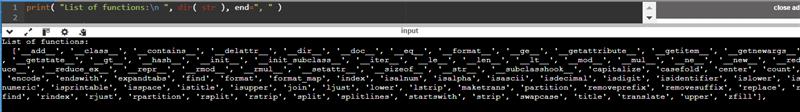
**2.4. Import all Names - From import \* Statement:**

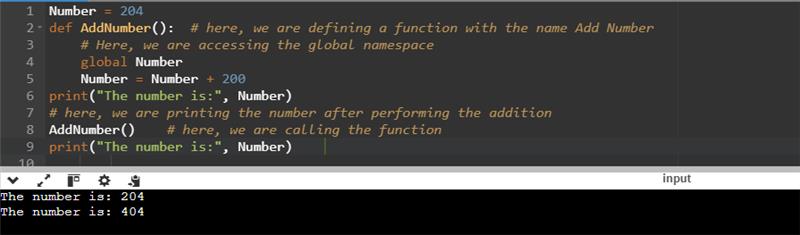
****

**2.5. Locating Path of Modules:**

****

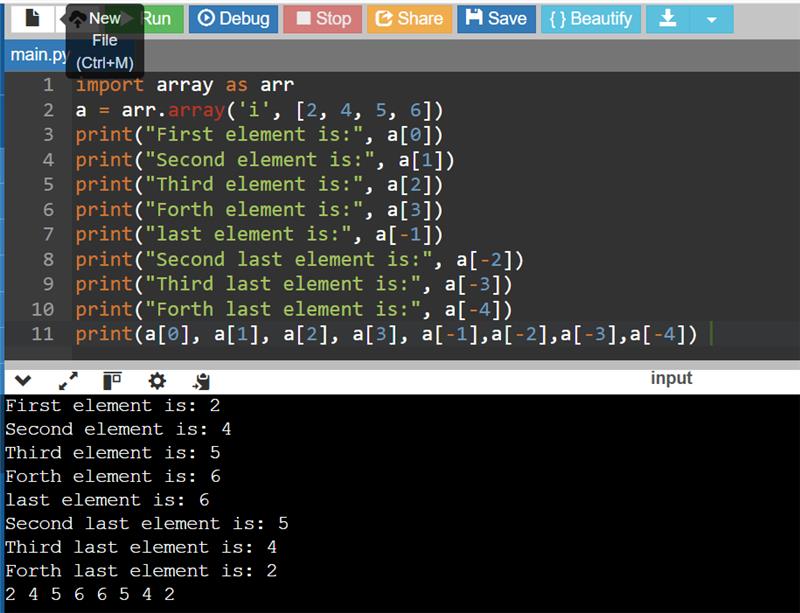
**2.6. The dir() Built-in Function:**

****

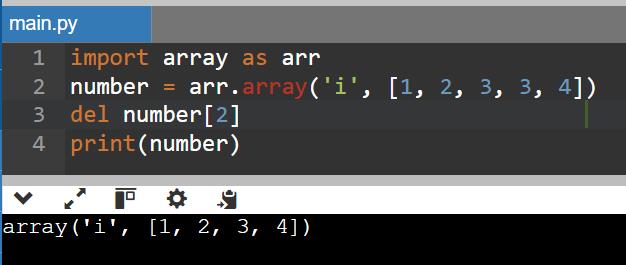
**2.7. Namespaces and Scoping:**

**3) PYTHON ARRAYS**

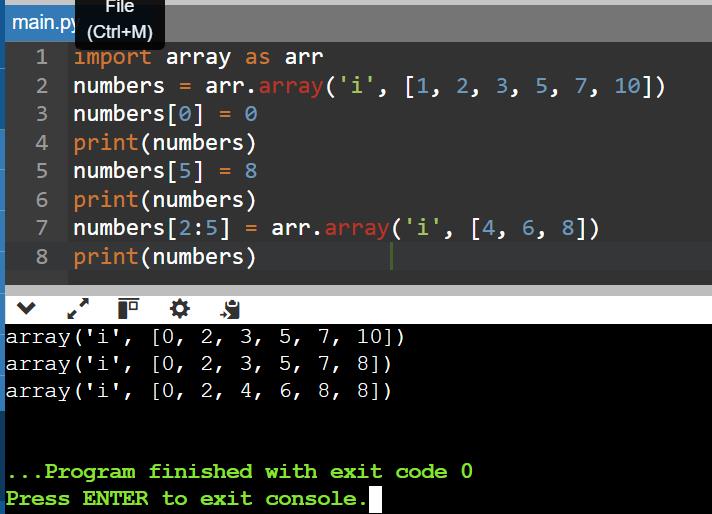
**3.1. Accessing array elements:**

****

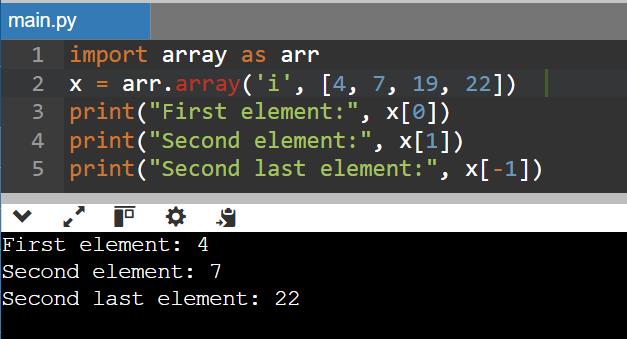
**3.2. Deleting the elements from Array**

****

**3.3. Adding or changing the elements in Array**

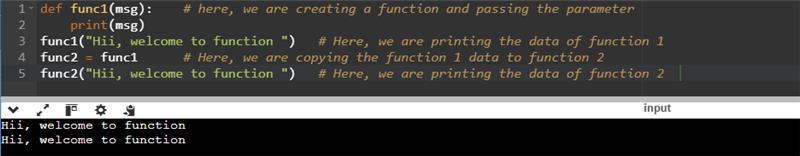
****

**3.4. To find the length of array**

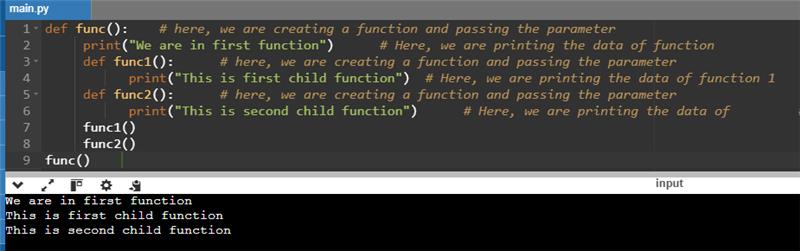
****

**4) PYTHON DECORATOR**

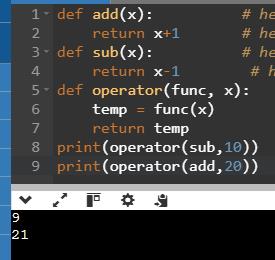
**4.1**

****

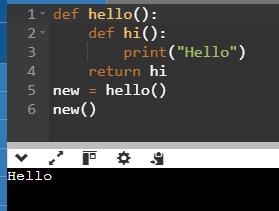
**4.2 Inner Function**

****

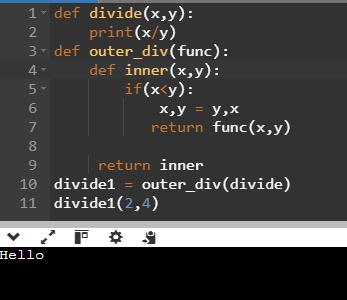
**4.3.**

****

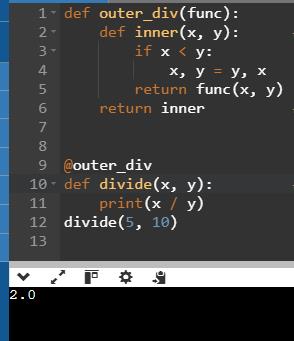
**4.4.**

****

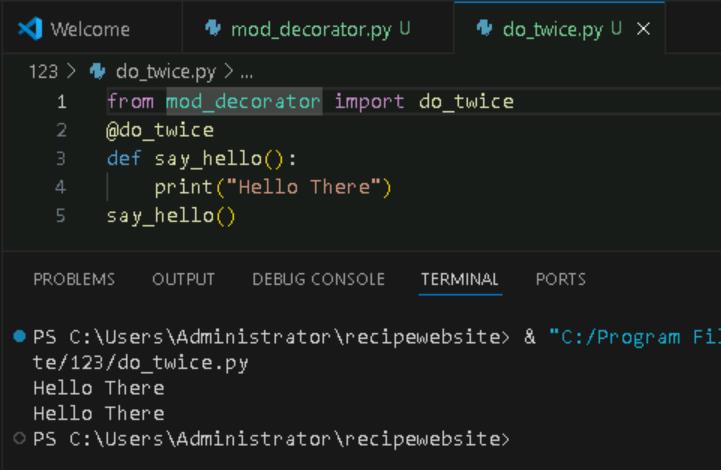
**4.5.Decorating functions with parameters:**

****

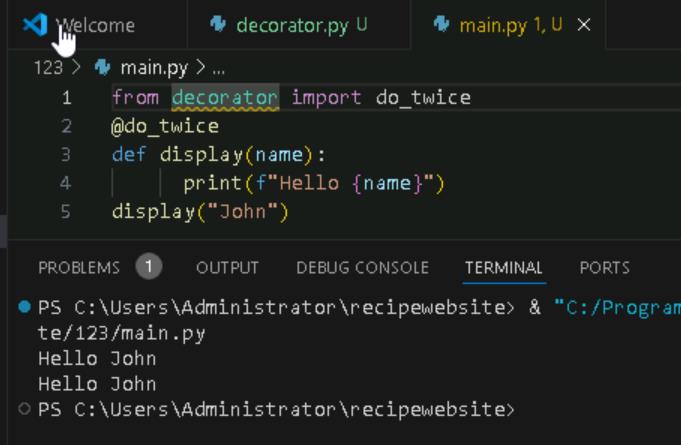
**4.6.Syntactic Decorator:**

****

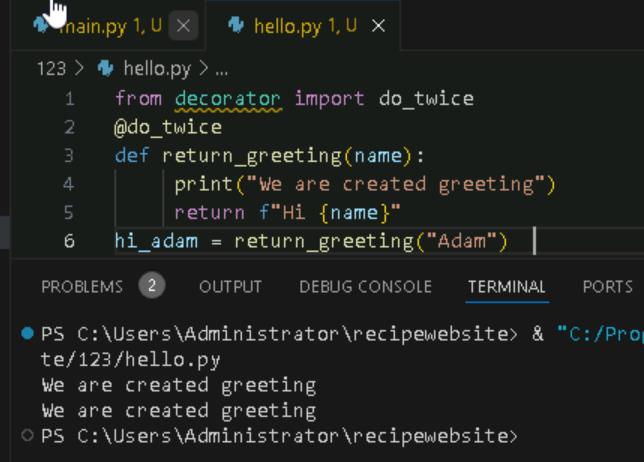
**4.7.Reusing Decorator**

****

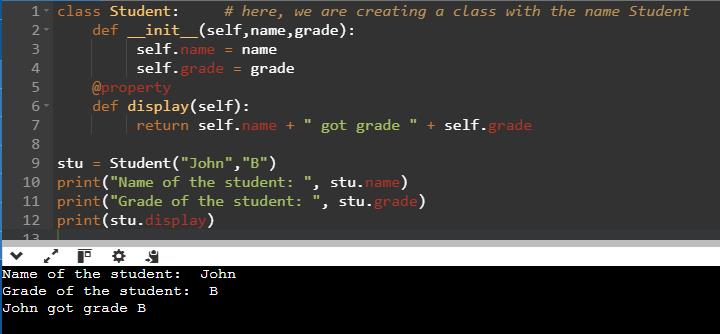
**4.8.Python Decorator with Argument**

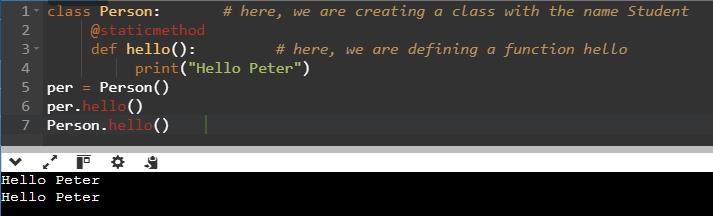
****

**4.9.Returning Values from Decorated Functions**

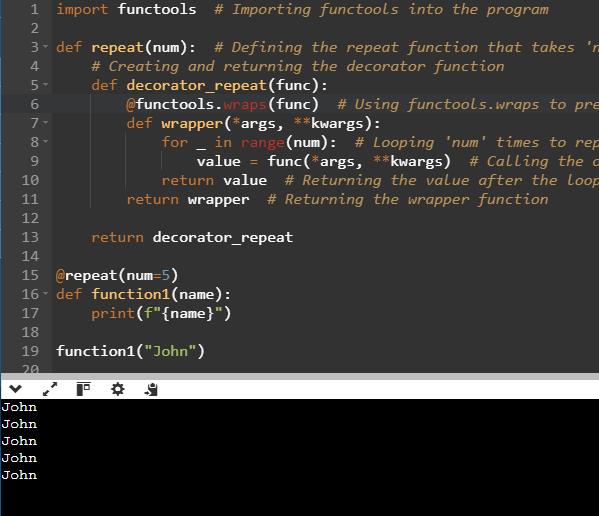
****

**4.10.Fancy Decorators**

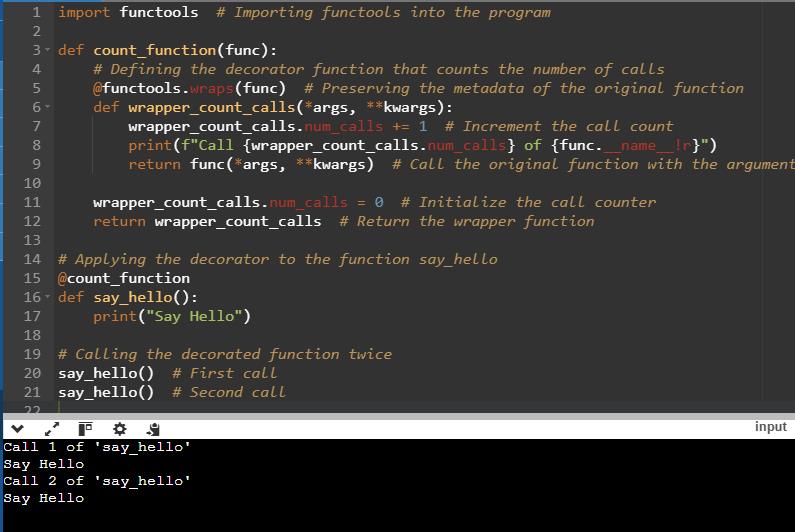
****

****

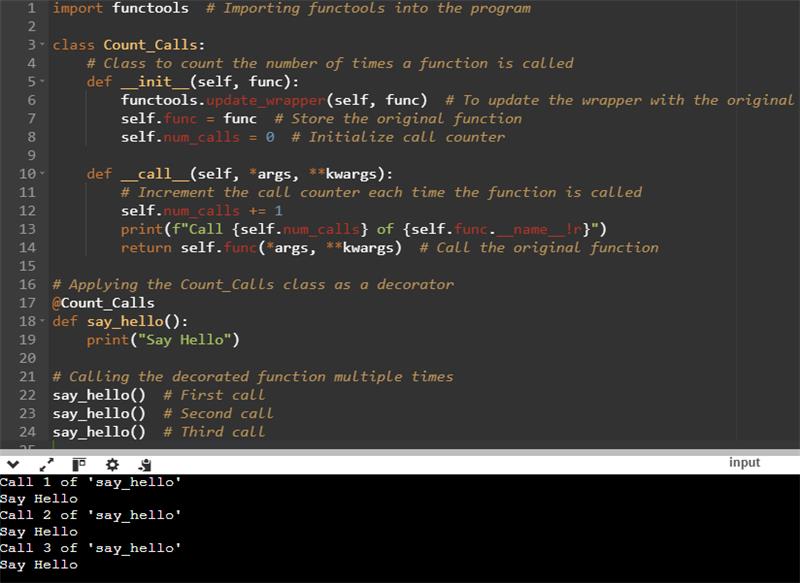
**4.11.Decorator with Arguments**

****

**4.12.Stateful Decorators**

****

**4.13.Classes as Decorators**

****