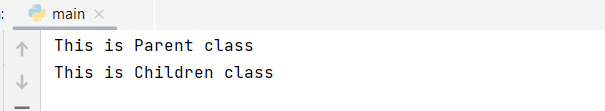
**Lab6B.py**

**Usecase:1**

****

****

Here we created two Classes with Person and Employee,  
Person is parent and Employee as child, Employee has extended from the Person

And Employee object has created and then functions are called.  
Employee has both functions ,parent and child functions, both are Inherited.

Employee object is created and both functions are called.

**Usecase2:**

**A screenshot of a computer program

Description automatically generated**

Explanation:

It’s a classic example, where child has been extended some parent properties, and why error is occurred here,

Here we created Parent Object and trying to call the children method which cannot be done.

Because when code execution is done ,whole code is compile and an Object is created for the Person. Address is assigned for it, now we are trying to call the child function which is not available in heap memory

**Usecase3:**

**A screenshot of a computer program

Description automatically generated**

**Explanation:-**

In this snippets we can see parent class will accepts two attributes, and a constructor has created for it and function is created and two parameters printed.

In Employee class extended from the parent level and here it does not have the any parameters,  
Child has extended it’s parent so it will get it’s parent information

While creating Child object so, we need to create the object and two parameters need to passed

Once passing the parameter parent data will be called in the child object.

**Usecase:4**

**A screenshot of a computer program

Description automatically generated**

**Explanation:-**

In Inheritance always child will have more priority than the parent class.

And Employee Class has 3 parameters and same params will be printed.

**Usecase:5**

**A screenshot of a computer program

Description automatically generated**

**Explanation:** Why Parent Data is printed, why not child data?

Here parent had 3 parameters and Child has 3 parameters.

As super parent is called by Person.\_\_init\_\_(self,emp\_name,emp\_age)

Person was called explicitly and then called function of that parent**.**

**Usecase :6**

****

**Explanation:**

The Employee class uses super().\_\_init\_\_(emp\_name, emp\_age) to call the constructor of the superclass (Person) and initialize the name and age attributes.

The display2 method in the Employee class uses super().display1() to call the display1 method of the superclass, which prints the name and age.

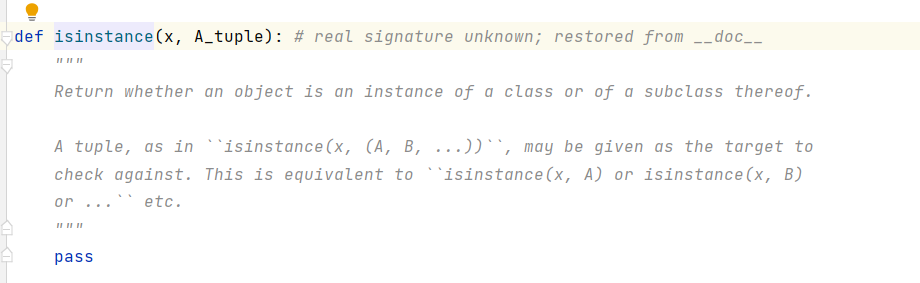
When you create an instance of the Employee class (emp), it inherits the attributes and methods from the Person class.

The display2 method is then called on the emp object, displaying both the salary and the information inherited from the Person superclass.

**Usecase:7**

**A screenshot of a computer program

Description automatically generated**

****

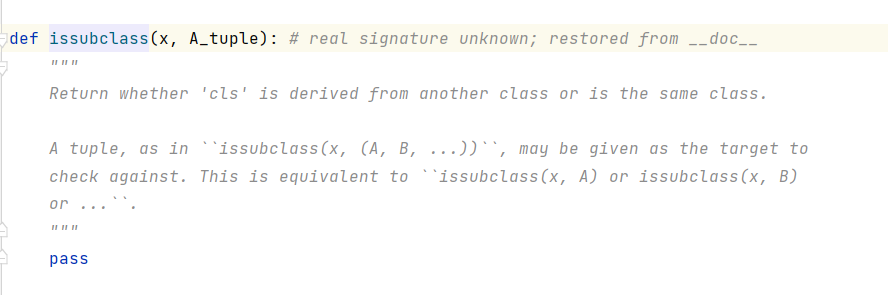
**Explanation:-**It will returns the is object is an instance of a class .

So it will checks about the current class is inherited from the parent or not, and returns Boolean value

**Usecase:8**

**A black rectangular object with white text

Description automatically generated**

****

**Pic:-Python Source code**

As above mentioned Child class and then followed the parent class and keep goes.

**Uecase:9**

****

**Explanation:**

It’s quite trick and easy, if we get to know how parent objects are called internally, this is how happened,  
Child will takes one param as input, because we have initiated the with one param.

And then parent was initiated with passing the child param twice and then we called the parent function to perform the operations

**Question:2**

*#Name:Pravalika Rao Chitneni  
#CompletionDate:24/11/2023 06.00AM  
  
  
#Here we have created Student Class,Constructor and toString methods are used.  
#Constructor used for the object initialisation and address is allocated to that object,so that during runtime code execution flow will be executed  
#Here Class Student will take three params as name,major,college and by using toSting method are printing the local variables  
#  
#Now we are creating another class Graduate class which will inherit the Student class data  
#Graduate class will takes another 5 parameter and it will do same execution  
# Graduate Class will takes student as param's and this is the way to declaration of the Inheritance  
# Class ChildrenClass:-GraduateStudent  
# Class ParentClass:-Student  
# ChildrenClass(ParentClass):  
# super().\_\_init\_\_(name, major, college) is the way to call the Parent # # Constructor from the Children.*

class Student:  
 def \_\_init\_\_(self, name, major,college):  
 self.name = name  
 self.college = college  
 self.major = major  
  
 def \_\_str\_\_(self):  
 return f"{self.name} is a/an {self.major} student at {self.college}"  
  
  
class GraduateStudent(Student):  
 def \_\_init\_\_(self, name, college, major, project, scholarship):  
 super().\_\_init\_\_(name, major, college)  
 self.project = project  
 self.scholarship = scholarship  
  
 def \_\_str\_\_(self):  
 return super().\_\_str\_\_() + f" working on project {self.project},receiving ${self.scholarship} as scholarship"  
  
  
*# Example usage:*student1 = Student("Steve", "English Student", "NEC")  
print(student1)  
  
grad\_student1 = GraduateStudent("Mery", "Mechanical Engineering", "MIT", "Dream Analysis",7800)  
print(grad\_student1)



