

Programación Orientada a Objetos. Taller 8 Python

David Fernando Cerón Quintero

1. Solo funciona para ob3 y ob4.

ob1: Porque, tanto la clase Ejecución1 y Números (Su clase padre) no definen el método sumar().

ob2: Aunque Operadores1 define el método sumar(), ni Ejecución2, ni Operadores1 u Operadores2 (Sus clases padres) tienen los atributos a y b .

ob3: Porque Ejecución3 tiene como padres las clases Números (Por ello, los atributos a y b) y Operadores1 (Por ende, el método, sumar()).

ob4: Porque Ejecución4 tiene como padres las clases Números (Por ello, los atributos a y b) y Operadores3 (Que tiene como clase padre a Operadores1).

2. Solo funciona para ob4.

ob1: Porque, tanto la clase Ejecución1 y Números (Su clase padre) no definen el método restar().

ob2: Aunque Operadores2 define el método restar(), ni Ejecución2, ni Operadores1 u Operadores2 (Sus clases padres) tienen los atributos a y b .

ob3: Aunque Ejecución3 tiene como padre la clase Números (Por ello, los atributos a y b), ni la clase Ejecución3, Números u Operadores1 (Sus clases padres) no definen el método restar().

ob4: Porque Ejecución4 tiene como padres las clases Números (Por ello, los atributos a y b) y Operadores3 (Que tiene como clase padre a Operadores2).

3. Solo funciona para ob3 y ob4.

ob1: Porque, tanto la clase Ejecución1 y Números (Su clase padre) no definen el método multiplicar().

ob2: Aunque Operadores1 define el método multiplicar(), ni Ejecución2, ni Operadores1 u Operadores2 (Sus clases padres) tienen los atributos a y b .

ob3: Porque Ejecución3 tiene como padres las clases Números (Por ello, los atributos a y b) y Operadores1 (Por ende, el método, multiplicar()).

ob4: Porque Ejecución4 tiene como padres las clases Números (Por ello, los atributos a y b) y Operadores3 (Que tiene como clase padre a Operadores1).

4. Solo funciona para ob4.

ob1: Porque, tanto la clase Ejecución1 y Números (Su clase padre) no definen el método dividir().

ob2: Aunque Operadores2 define el método dividir(), ni Ejecución2, ni Operadores1 u Operadores2 (Sus clases padres) tienen los atributos a y b .

ob3: Aunque Ejecución3 tiene como padre la clase Números (Por ello, los atributos a y b), ni la clase Ejecución3, Números u Operadores1 (Sus clases padres) no definen el método dividir().

ob4: Porque Ejecución4 tiene como padres las clases Números (Por ello, los atributos a y b) y Operadores3 (Que tiene como clase padre a Operadores2).

5. Dependiendo de la connotación de la pregunta, podría variar la respuesta. Si esta hace referencia a que cualquier objeto de esta clase no herede la forma de otra clase (Es decir, la clase original no tenga clases padre), serían las clases Números, Operaciones1 y Operaciones2. Si hace referencia a que no puedan existir objetos instanciados con otras clases distintas a la mencionada (Es decir, no existen otras clases que tengan a esta como clase padre), serían Ejecución1, Ejecución2, Ejecución3 y Ejecución4. Si hace referencia a que no pueda existir un polimorfismo con un objeto instanciado con esta clase, ninguna.