

HOUSEHOLD APPLIANCES
-name: str -unit: str -power: int -warranty_validity: int -hours_used: int
+power_consumption () +get_warranty ()

```
class Household_Appliance:
    def __init__(self, name: str, unit: str, power: float, warranty_validity: int, hours_used: float):
        self.name = name
        self.unit = unit
        self.power = power
        self.warranty_validity = warranty_validity
        self.hours_used = hours_used

    def power_consumption(self):
        print(f"{self.name} power consumption is: {self.power * self.hours_used}")

    def get_warranty(self):
        print(f"{self.name} has {self.warranty_validity} year/s warranty!")
```

ANIMALS
-name: str -species: str -color: int
+identify_breed +pet_color

```
class Animals:
    def __init__(self, name: str, species: str, color: str):
        self.name = name
        self.species = species
        self.color = color

    def identify_breed(self):
        print(f"{self.name} is a {self.species}")

    def pet_color(self):
        print(f"{self.name} is color {self.color}!")
```

PRODUCTS
-name: str -price: float -discount: bool
+prod_identification +prod_disc

```
class Products:
    def __init__(self, name: str, price: float, discount: bool):
        self.name = name
        self.price = price
        self.discount = discount

    def prod_identification(self):
        print(f"Product: {self.name}")
        print(f"Price: {self.price}")

    def prod_disc(self):
        if self.discount:
            print(f"{self.name} is discounted for {self.discount}!")
        else:
            print(f"{self.name} is not discounted!")
```

OUTPUT:

```
Desktop = Household_Appliance('Desktop', 'ryzen', 100.0, 2, 5.0)
Desktop.power_consumption()
Desktop.get_warranty()

stand_fan = Household_Appliance('Stand fan', 'tough mama', 13, 1, 12)
stand_fan.power_consumption()
stand_fan.get_warranty()

speaker = Household_Appliance('Speaker', 'Panasonic', 20, 4, 2)
speaker.power_consumption()
speaker.get_warranty()

print('-----')

Dog = Animals('Beerus', 'Mini Pinscher', 'Brown')
Dog.identify_breed()
Dog.pet_color()

Dog = Animals('Pencil', 'Mini Pinscher', 'Black')
Dog.identify_breed()
Dog.pet_color()

Cat = Animals('Chingkay', 'Puspin', 'White-Orange')
Cat.identify_breed()
Cat.pet_color()
print('-----')
```

```
Cat = Animals('Chingkay', 'Puspin', 'White-Orange')
Cat.identify_breed()
Cat.pet_color()
print('-----')

Monitor = Products('Asus Monitor', '12,500', '12%')
Monitor.prod_identification()
Monitor.prod_disc()

Phone = Products('Redmi Phone', '17 500', False)
Phone.prod_identification()
Phone.prod_disc()

Helmet = Products('EVO Helmet', '4 999', '30%')
Helmet.prod_identification()
Helmet.prod_disc()
```

```
Desktop power consumption is: 500.0
Desktop has 2 year/s warranty!
Stand fan power consumption is: 156
Stand fan has 1 year/s warranty!
Speaker power consumption is: 40
Speaker has 4 year/s warranty!
```

```
-----
Beerus is a Mini Pinscher
Beerus is color Brown!
Pencil is a Mini Pinscher
Pencil is color Black!
Chingkay is a Puspin
Chingkay is color White-Orange!
```

```
-----
Product: Asus Monitor
Price: 12,500
Asus Monitor is discounted for 12%!
Product: Redmi Phone
Price: 17 500
Redmi Phone is not discounted!
Product: EVO Helmet
Price: 4 999
EVO Helmet is discounted for 30%!
```