

```
from datetime import datetime
##import datetime
```

```
class CarRental:
    def __init__(self, stock=0):          ## -> None:
        self.stock = stock
        self.rental_time = 0
        self.rate = 0
        self.basis = 'none'

    def displaystock(self):
        print(f"There are {self.stock} cars available")

    def rent_hourly(self, n):
        if n <= 0:
            print("Num of cars must be positive")
        elif n > self.stock:
            print(f"We can not rent {n} cars thre are only {self.stock} left")
        else:
            self.rental_time = datetime.now()
            self.rate = 20
            self.basis = 'hour'
            print(f"You have rented {n} cars at an agreed upon rate of {self.rate}
per {self.basis} starting at {self.rental_time}")
            self.stock -= n
            return True
        return False

    def rent_daily(self, n):
        if n <= 0:
            print("Num of cars must be positive")
        elif n > self.stock:
            print(f"We can not rent {n} cars thre are only {self.stock} left")
        else:
            self.rental_time = datetime.now()
            self.rate = 100
            self.basis = 'daily'
            print(f"You have rented {n} cars at an agreed upon rate of {self.rate}
per {self.basis} starting at {self.rental_time}")
            self.stock -= n
            return True
        return False

    def rent_weekly(self, n):
        if n <= 0:
            print("Num of cars must be positive")
        elif n > self.stock:
            print(f"We can not rent {n} cars thre are only {self.stock} left")
        else:
```

```

        self.rental_time = datetime.now()
        self.rate = 750
        self.basis = 'weekly'
        print(f"You have rented {n} cars at an agreed upon rate of {self.rate}
per {self.basis} starting at {self.rental_time}")
        self.stock -= n
        return True
    return False

def returncar(self, n):
    bill = 0

    if n <= 0:
        print(f"We can not return {n} cars")
        return False

    if self.rental_time == 0:
        print("The rental time was never set")
        return False

    self.stock += n
    now = datetime.now()

    if(self.basis == 'hour'):
        bill = (now.hour - self.rental_time.hour) * self.rate * n
    elif(self.basis == 'daily'):
        bill = (now.day - self.rental_time.day) * self.rate * n
    elif(self.basis == 'weekly'):
        bill = (now.day - self.rental_time.day)/7 * self.rate * n
    else:
        print(f"The basis was not set correctly it is currently {self.basis}")
        return False

    print(f"The total of your bill is {bill} this is from renting {n} cars at
{self.rate} on a {self.basis} rate")
    print("Thank you please come again")
    return True

```

```

class Customer:
    def __init__(self, name):          ## -> None:
        self.name = name
        self.num_cars = 0
        ##self.bill = 0               ## save this for a later date to update with the
ability to pay at a later time

    def request(self, n):
        if n <= 0:
            print(f"You ({self.name}) are trying to request {n} cars which we can
not allow")

```

```
    else:
        self.num_cars = n

def car_return(self,n,):
    if self and n:
        self.num_cars -= n
        ##self.bill = bill
        print(f"You have returned {self.num_cars}, thank you come again.")
    else:
        print("You aren't trying to return anything.")
```