

13.1. Тээврийн хэрэгсэл vehicle классыг дээд хурд болон явсан зам атрибуттай үүсгэх.

```
In [1]: class Vehicle:
        def __init__(self, max_speed, mileage):
            self.max_speed = max_speed
            self.mileage = mileage

        modelX = Vehicle(240, 18)
        print(modelX.max_speed, modelX.mileage)
```

240 18

13.2. Өмнө үүсгэсэн классын Чайлд класс үүсгэхдээ тээврийн хэрэгсэл vehicle классаас удамшсан байх ёстой.

```
In [2]: class Bus(Vehicle):
        pass
```

13.3. Өмнө үүсгэсэн классын Bus нэртэй чайлд класс үүсгэхдээ тээврийн хэрэгсэл vehicle классаас удамшсан байх ёстой ба суудлын тооны тогтмол утгыг 50 байхаар өгнө.

```
In [3]: class Vehicle:
        def __init__(self, name, max_speed, mileage):
            self.name = name
            self.max_speed = max_speed
            self.mileage = mileage

        class Bus(Vehicle):
            pass

        School_bus = Bus("School Volvo", 180, 12)
```

13.4. Өмнө үүсгэсэн классын Тээврийн хэрэгсэл vehicle классын өнгө буюу color атрибутын утга нь тогтмол утгаараа цагаан өнгө байхаар сонгоно.

```
In [4]: class Vehicle:
    def __init__(self, name, max_speed, mileage):
        self.name = name
        self.max_speed = max_speed
        self.mileage = mileage

    def seating_capacity(self, capacity):
        return f"The seating capacity of a {self.name} is {capacity} passengers"

class Bus(Vehicle):
    # assign default value to capacity
    def seating_capacity(self, capacity=50):
        return super().seating_capacity(capacity=50)

School_bus = Bus("School Volvo", 180, 12)
```

13.5. Өмнө үүсгэсэн классын Bus нэртэй чайд класс үүсгэхдээ тээврийн хэрэгсэл vehicle классаас удамшсан байх ёстой ба суудлын тоо 50-г нэг зорчигчийн тээврийн хөлс 100 доллар байхаар тооцож хүн тус бүрийн төлбөр дээр 10 хувийн татвар нэмнэ. Төлбөрийн мэдээллийг fare() нэртэй функц дотор хуулж, өгнө.

```
In [5]: class Vehicle:
    color = "White"

    def __init__(self, name, max_speed, mileage):
        self.name = name
        self.max_speed = max_speed
        self.mileage = mileage

class Bus(Vehicle):
    pass

class Car(Vehicle):
    pass

School_bus = Bus("School Volvo", 180, 12)
print(School_bus.color, School_bus.name, "Speed:", School_bus.max_speed, "Mileage:", School_bus.mileage)

car = Car("Audi Q5", 240, 18)
print(car.color, car.name, "Speed:", car.max_speed, "Mileage:", car.mileage)

White School Volvo Speed: 180 Mileage: 12
White Audi Q5 Speed: 240 Mileage: 18
```

13.6. Өмнө үүсгэсэн классын School_bus хувьсагч нь vehicle болон bus гэсэн классаас давхар удамшсан бол түүнийг үүсгэнэ.

```
In [6]: class Vehicle:
        def __init__(self, name, mileage, capacity):
            self.name = name
            self.mileage = mileage
            self.capacity = capacity

        def fare(self):
            return self.capacity * 100

class Bus(Vehicle):
    def fare(self):
        amount = super().fare()
        amount += amount * 10 / 100
        return amount

School_bus = Bus("School Volvo", 12, 50)
print("Total Bus fare is:", School_bus.fare())
```

Total Bus fare is: 5500.0

13.7. Оюутан student нэртэй класс үүсгэнэ. Энэ класс нь оюутны код, оюутны нэр, хичээлийн код болон дүн нэртэй атрибуттай байна.

```
In [7]: class student:

        def __init__(self, code:str = None, name:str = None, lesson_code:str = None,
            self.student_code = code
            self.student_name = name
            self.lesson_code = lesson_code
            self.school_result = result
```

13.8. Хичээл нэртэй классыг хичээлийн нэр, кредит, заадаг багшийн код, заадаг багшийн нэр атрибуттайгаар үүсгэнэ. Мөн энэ класс нь багш нэртэй классаас удамшсан байна. Багш класс нь багшийн нэр, багшийн код, багшийн зэрэг, цол, багшийн ажилласан жил, багшийн харъяалагдах тэнхим, багшийн харъяалагдах сургууль атрибуттайгаар үүсгэнэ.

```
In [8]: class Teacher:

    def __init__(self, teacher_name:str=None, teacher_code:str=None, teacher_deg
        teacher_worked_year:int=None, teacher_department:str=None):
        self.teacher_name = teacher_name
        self.teacher_code = teacher_code
        self.teacher_degree = teacher_degree
        self.teacher_worked_year = teacher_worked_year
        self.teacher_department = teacher_department
        self.teacher_school = teacher_school

class Lesson(Teacher):

    def __init__(self, lesson_name:str=None, kredit_score:int=None, teacher_code:
        super().__init__(teacher_name = teacher_name, teacher_code = teacher_code)
        self.lesson_name = lesson_name
        self.kredit_score = kredit_score

lesson = Lesson(lesson_name='Algorithms', kredit_score=3, teacher_code='P.CS034',
print(lesson.teacher_name)
```

Teacher

13.9. Өмнө үүсгэсэн оюутан классыг хичээл нэртэй классаас удамшсан байхаар үүсгэнэ.

```
In [9]: class Teacher:

    def __init__(self, teacher_name:str=None, teacher_code:str=None, teacher_deg
        teacher_worked_year:int=None, teacher_department:str=None):
        self.teacher_name = teacher_name
        self.teacher_code = teacher_code
        self.teacher_degree = teacher_degree
        self.teacher_worked_year = teacher_worked_year
        self.teacher_department = teacher_department
        self.teacher_school = teacher_school

class Lesson(Teacher):

    def __init__(self, lesson_name:str=None, lesson_code:str=None, kredit_score:i
        super().__init__(teacher_name = teacher_name, teacher_code = teacher_code)
        self.lesson_name = lesson_name
        self.kredit_score = kredit_score
        self.lesson_code = lesson_code

class Student(Lesson):

    def __init__(self, code:str = None, name:str = None, lesson_code:str = None,
        super().__init__(lesson_code=lesson_code, teacher_code=teacher_code)
        self.student_code = code
        self.student_name = name
        self.school_result = result

student = Student(result=96, lesson_code='P.IT101', name='Name', code='Code', tea

print(student.teacher_code)
```

P.CS034