

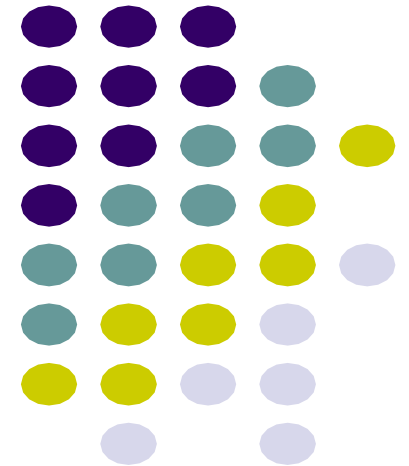
ШИНЖЛЭХ УХААН ТЕХНОЛОГИЙН ИХ СУРГУУЛЬ  
МЭДЭЭЛЭЛ, ХОЛБООНЫ ТЕХНОЛОГИЙН СУРГУУЛЬ

F.CS202  
ОБЪЕКТ ХАНДЛАГАТ ПРОГРАМЧЛАЛ

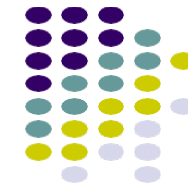
Лекц №3

**Битүүмжлэл ба классын бүрэлдэхүүн**

2021 он



# Агуулга



- ★ Өгөгдөл/атрибут
- ★ Үйлдэл/метод
- ★ Асуух үйлдэл
- ★ Өөрчлөх үйлдэл
- ★ ЮМЛ дүрслэл



# Өмнөх лекцээр

# Объект хандлагат програмчлал



- ★ Чухал ойлголт:
  - ★ Класс
  - ★ Объект – биет болон биет бус
- ★ Объектуудын харилцаанаас тогтоно

# Оюутны дотуур байрны бүртгэлийн програм



Давхрын дугаар  
Өрөөний тоо  
өрөөнийТооӨөрчлөх()

**Давхар**

Өгөгдөл/атрибут

Үйлдэл/Метод

Класс

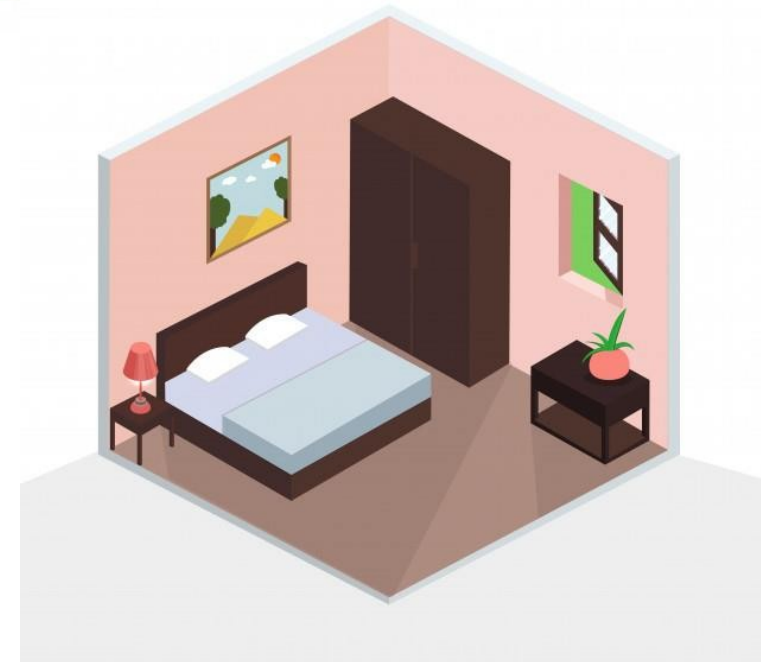


**Оюутан**

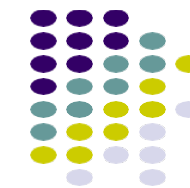
Нэр  
Хүйс  
Төрсөн огноо  
Утасны дугаар  
Хаяг  
утасӨөрчлөх()  
хаягХарах()

Өрөөний дугаар  
Орны тоо  
Мкв  
Хүнтэй эсэх  
орныТооӨөрчлөх()  
мквХарах()

**Өрөө**



# Дугуйн дэлгүүрийн борлуулалтын бүртгэлийн програм



Дугуйны үйлдвэр  
Дугуйны хэмжээ  
Зориулалт  
Өнгө  
Үнэ  
үнэӨөрчлөх()  
үнэХарах()

← Өгөгдөл/атрибут

← Үйлдэл/Метод

Дугуй

← Класс



Үйлчлүүлэгч

нэр  
утас  
хаяг  
утасӨөрчлөх()  
хаягХарах()

← Класс

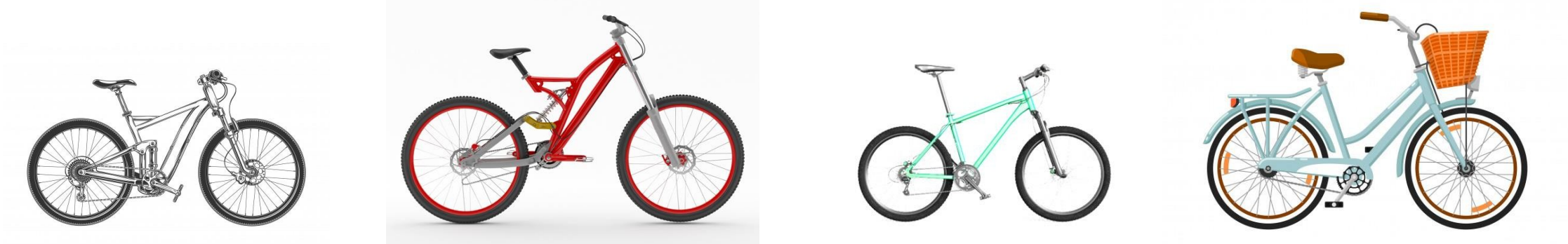
← Өгөгдөл/атрибут

← Үйлдэл/Метод

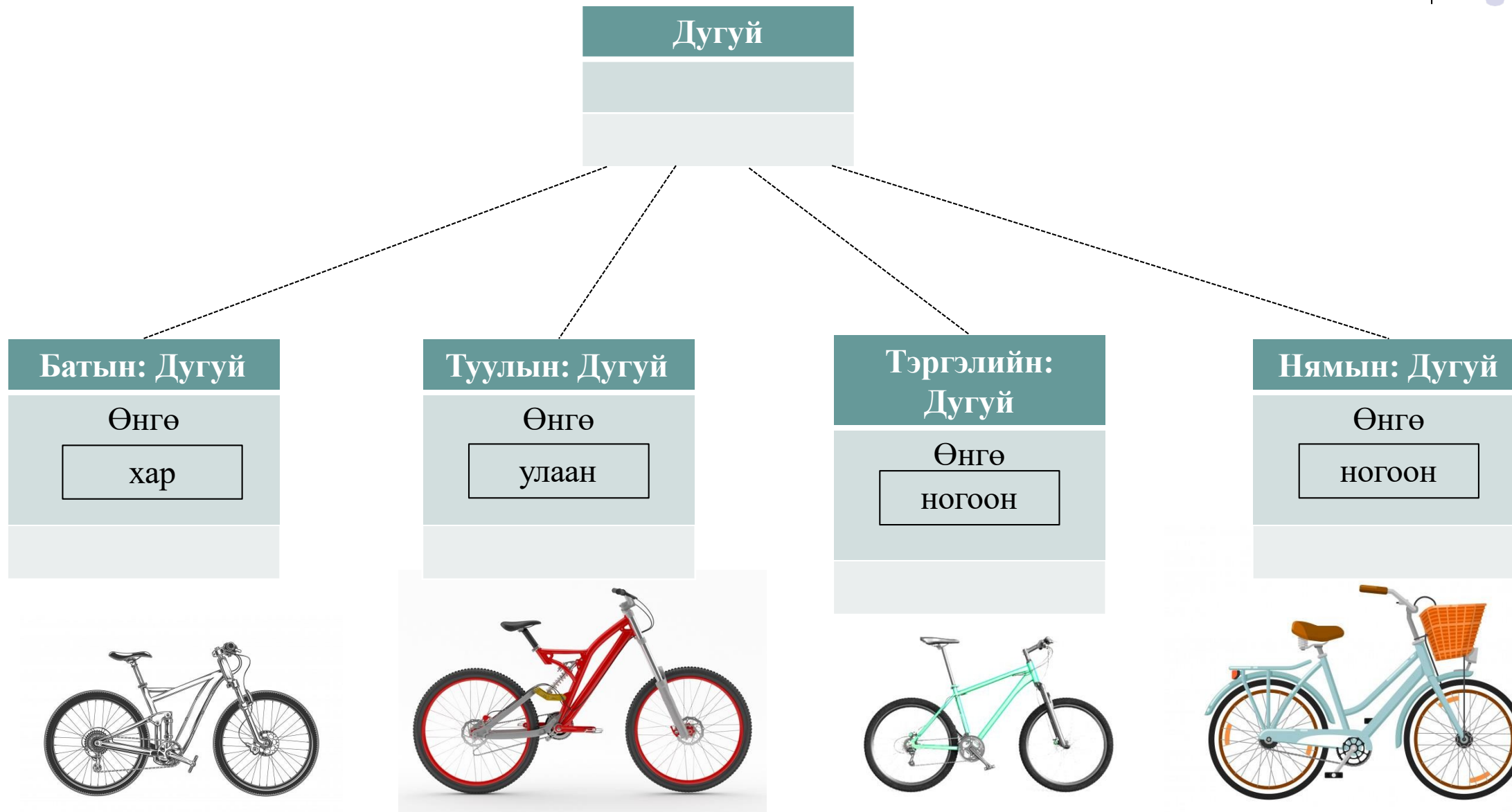


# Класс ба объект

- ★ Програмд объектыг үүсгэх зааврыг бичнэ.
- ★ Энэ тодорхойлолтыг **Класс** гэж нэрлэнэ.
- ★ Объект — классын тохиолдол/инстанс
- ★ Классыг нэгэнт тодорхойлсон бол үүнээс хэдэн ч инстанс(объект) үүсгэж болно.



# Класс ба объект





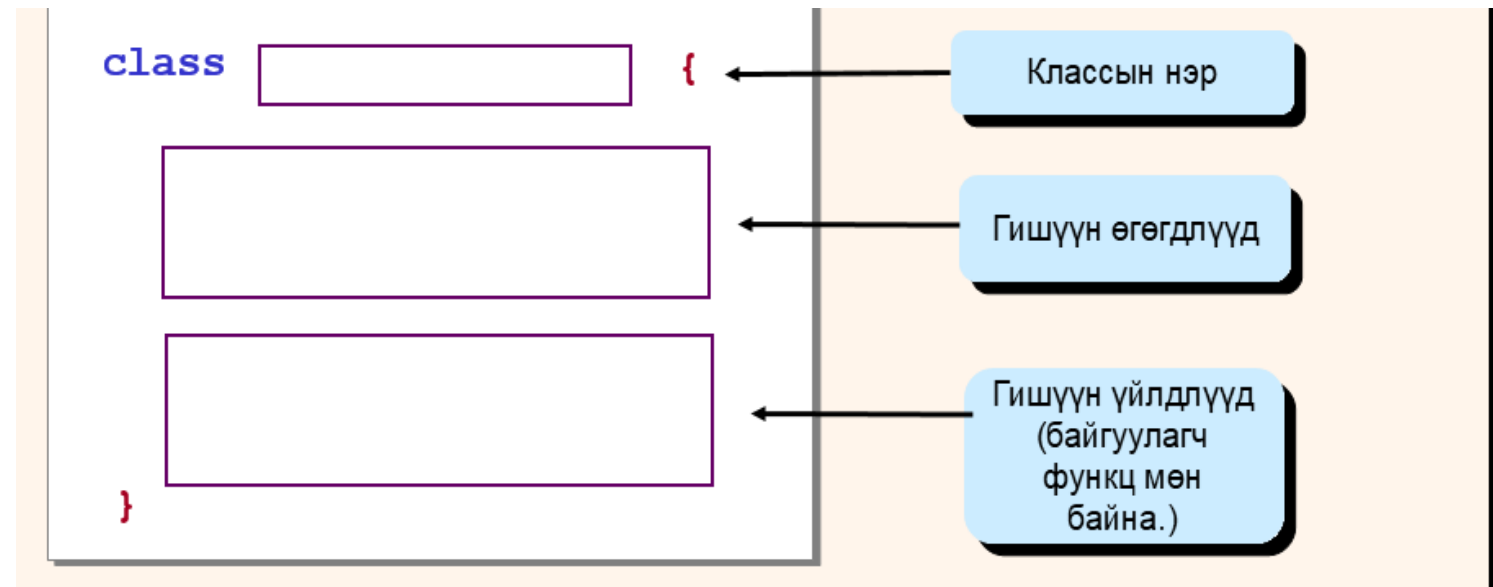


## Классын бүрэлдэхүүн хэсгүүд



# Классын бүрэлдэхүүн хэсгүүд

- ★ Өгөгдөл/атрибут – гишүүн, атрибут
- ★ Үйлдэл/метод - харьцаа
- ★ Байгуулагч функц
- ★ Устгагч функц



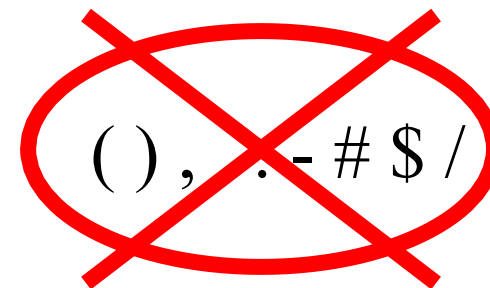
# Классыг нэрлэх



★ Классын нэр PascalCase хэлбэрээр байна. Хоосон зай зөвшөөрөхгүй.

★ BookOrder, ScoreWriter, Student, TicTacToe

★ Тусгай тэмдэгтүүд ~~г~~ болохгүй.



★ Үг товчлохоос зайлсхийх.

FlightSegment



FltSgmnt

DepositAccount

DpstAccnt

★ Класс нь нэр үг эсвэл хэлц нэр үг байна.

★ Person, Money, BankAccount гэх мэт.

# Өгөгдөл



- ★ Ядаж нэг ялгах шинж илэрхийлсэн өгөгдөл (дахин давтагдашгүй),
  - ★ Төлөв илэрхийлсэн өгөгдлүүд,
  - ★ Шинж чанар илэрхийлсэн өгөгдлүүд,
  - ★ Хадгалж буй утга, өгөгдлүүд.
- 
- ★ Өгөгдөл нь camelCase хэлбэртэй эхлээд жижиг үсгээр эхэлж дараа нь том жижиг үсэг холилдон орно.
  - ★ Өгөгдлийн нэр ихэвчлэн нэр үг эсвэл хэлц нэр үг байдаг.

# Гишүүн өгөгдлийн зарлагаа



```
<modifiers> <data type> <name> ;
```

Харагдалтын  
тодорхойлогч



`private`

Өгөгдлийн төрөл



`String`

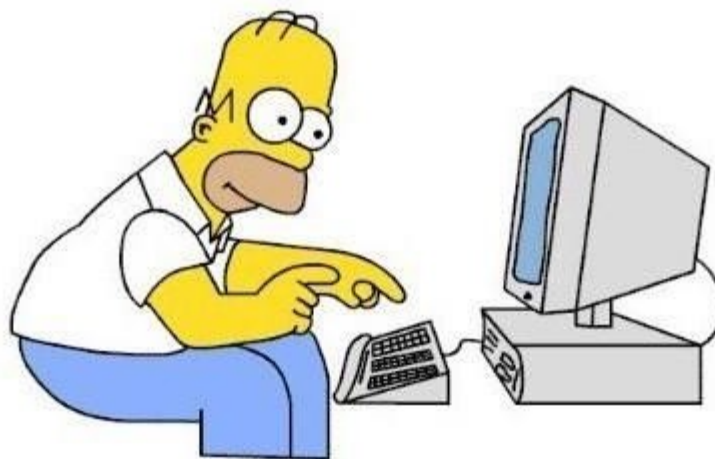
нэр



`ownerName ;`



Ямар ч тэнэг компьютер ойлгохоор  
код бичиж чаддаг. Сайн програмист  
л харин ХҮН ойлгох код бичиж чаддаг.



-Martin Fowler-

# Үйлдэл

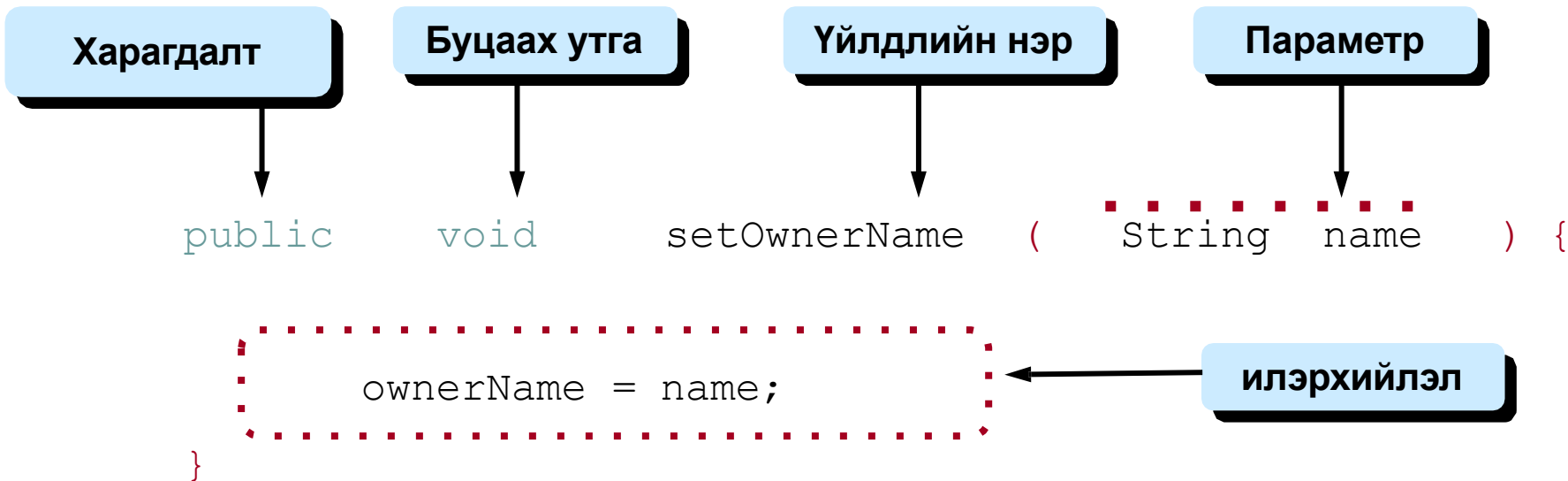


- ★ Үйлдэл нь тухайн классын хүрээний функц юм.
- ★ Үйлдлийн нэрийг camelCase хэлбэрээр тодорхойлдог ба үйлдлийн нэр нь ихэвчлэн үйл үг эсвэл хэлц үйл үг байдаг.
- ★ Параметрийн нэр нь мөн camelCase хэлбэртэй байх ба тэд үйлдэл рүү ямар нэгэн зүйл дамжуулахыг заадаг тул ихэвчлэн нэр үг эсвэл хэлц нэр үг байдаг.
- ★ Дотоод үйлдэл нь зөвхөн класс дотроосоо дуудаж ашиглах үйлдэл байна. Дотоод үйлдэл **private** байна.
- ★ Гадаад үйлдэл нь бусад классаас дуудах үйлдэл юм. Гадаад үйлдэл ихэвчлэн **public** байна.

# Гишүүн үйлдлийн зарлагаа



```
<modifier> <return type> <method name> ( <parameters> ) {  
    <statements>  
}
```



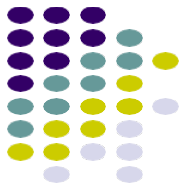


# Мэдээллийг нууцлах ба харагдалтыг тодорхойлогч



- ★ public болон private тодорхойлогчид нь гишүүн өгөгдөл болон методуудын хандалтыг илэрхийлдэг.
- ★ Классын бүрэлдэхүүн нь (гишүүн өгөгдөл , метод) private гэж зарлагдсан бол клиент класс эдгээрт хандаж чадахгүй.
- ★ Харин public гэж зарлагдсан бол клиент классаас хандах боломжтой болно.
- ★ Классын дотоод үйл ажиллагааг private-р зарласнаар клиентүүдээс нуух боломжтой. Үүнийг мэдээллийн нууцлалт гэнэ.

# Хандалтын жишээ



```
...  
Service obj = new Service();  
obj.memberOne = 10;  
obj.memberTwo = 20;  
obj.doOne();  
obj.doTwo();
```



Клиент

```
class Service {  
    public int memberOne;  
    private int memberTwo;  
    public void doOne() {  
        ...  
    }  
    private void doTwo() {  
        ...  
    }  
}
```

Сервис

# Гишүүн өгөгдөл `private` байх ёстой



- ★ Гишүүн өгөгдлүүд нь классын хэрэгжүүлэлтийн дэлгэрэнгүй нь учраас үүнийг клиентэнд харагдахгүй байлгах хэрэгтэй. Иймд гишүүн өгөгдлүүдийг `private` –р зарлана.
- ★ Exception: Хэрвээ тогтмолыг гадаад методуудаас шууд ашиглах шаардлагатай бол тэдгээрийг `public` харагдалттай зарлана.

# Харагдалтын тодорхойлогчийг зөв сонгох нь



- ★ Гишүүн өгөгдөл болон методын харагдалтыг зөв тодорхойлох зөвлөмж:
  - ★ Класс болон инстансын хувьсагчуудыг хувийн харагдалттайгаар зарлана.
  - ★ Класс болон инстансын методууд нь зөвхөн тухайн классынхаа бусад методуудад л ашиглагддаг бол `private` харагдалттай байна.
  - ★ Клиент програмд ашиглагддаг тогтмолыг нийтийн харагдалттай зарлана. Харин зөвхөн класс дотроо л ашиглагдах бол `private` байна.

# Харагдалтыг диаграм дээр тэмдэглэх



AccountVer2
<ul style="list-style-type: none"><li>- balance</li><li>- ownerName</li></ul>
<ul style="list-style-type: none"><li>+ AccountVer2( String, double )</li><li>+ add( double )</li><li>+ deduct( double )</li><li>+ getCurrentBalance( )</li><li>+ getOwnerName( )</li><li>+ setOwnerName( String )</li></ul>

public – нэмэх тэмдэг(+)

private – хасах тэмдэг(-)

# Объект зарлах



Классын нэр  
Классыг үүсгэсэн  
байх ёстой

Объектын нэр  
Объектын зарлагаа

Account  
Student  
Vehicle

customer;  
jan, jim, jon;  
car1, car2;

# Объект үүсгэх

**<объектын нэр> = new <Классын нэр> (<аргумент>)**

Объектын нэр  
Үүсгэж буй объект.

Классын нэр  
Энэ классын инстанс  
үүснэ.

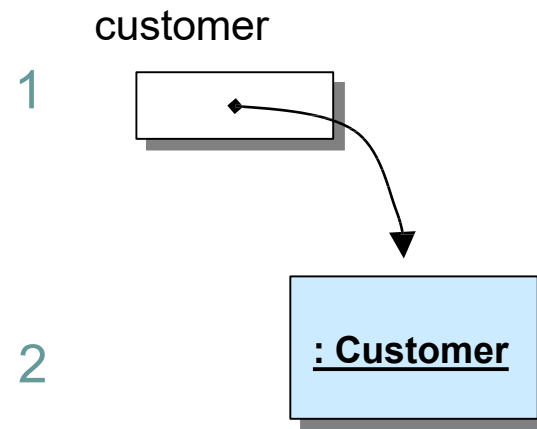
Аргумент

```
customer = new Customer( );  
jon      = new Student("John Java");  
car1     = new Vehicle( );
```

# Зарлах Ба Үүсгэхийн ялгаа



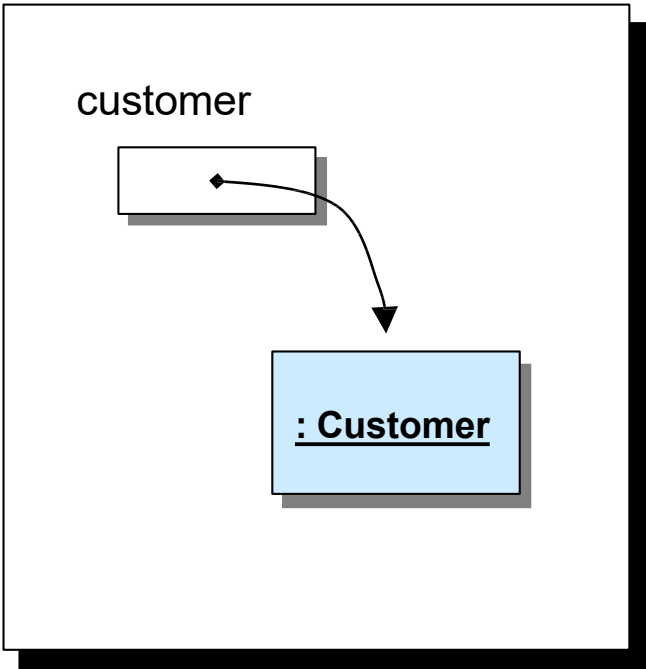
```
1  Customer  customer;  
2  customer  =  new  Customer( );
```



1. **customer** нэр зарлагдахад санах ойд зай авна.

2. **Customer** объектыг үүсгэхэд **customer** нэр нь үүнтэй холбогдоно.

# Санах ойн төлөв ба програм



Санах ойн төлөв



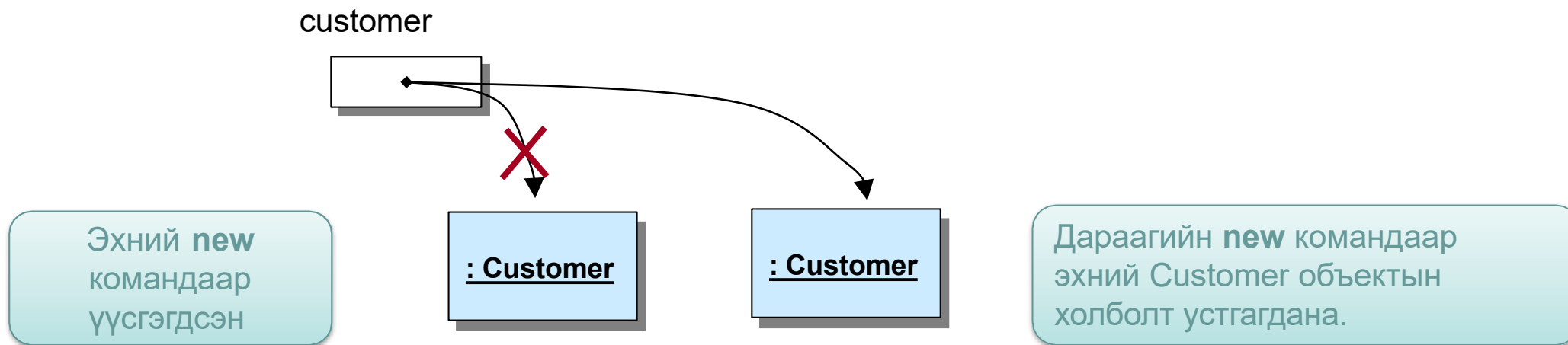
Програмын диаграм



# Нэр ба объект



```
Customer    customer;  
  
customer    = new Customer( );  
  
customer    = new Customer( );
```



# Класс, объектын жишээ



зарлах

```
class Cat {  
    public int id;  
    private string name;  
}
```

объект үүсгэх, хандах

```
class ZooPark {  
    public feedCat() {  
        Cat masha = new Cat();  
        masha.id = 5;  
        masha.name = "masha";  
        int i = masha.id;  
        string ner = masha.name;  
    }  
}
```

# Эхний жишээ: Bicycle классыг ашиглах



```
class BicycleRegistration {  
    public static void main(String[] args) {  
        Bicycle bike1, bike2;  
        String owner1, owner2;  
  
        bike1 = new Bicycle( );    //bike1 объект үүсгэх  
        bike1.setOwnerName("Adam Smith");  
  
        bike2 = new Bicycle( );    //bike2 объектыг үүсгэх  
        bike2.setOwnerName("Ben Jones");  
  
        owner1 = bike1.getOwnerName( ); //мэдээллийг хэвлэх  
        owner2 = bike2.getOwnerName( );  
  
        System.out.println(owner1 + " дугуйтай.");  
        System.out.println(owner2 + " мөн дугуйтай.");  
    }  
}
```

# Bicycle классын тодорхойлолт



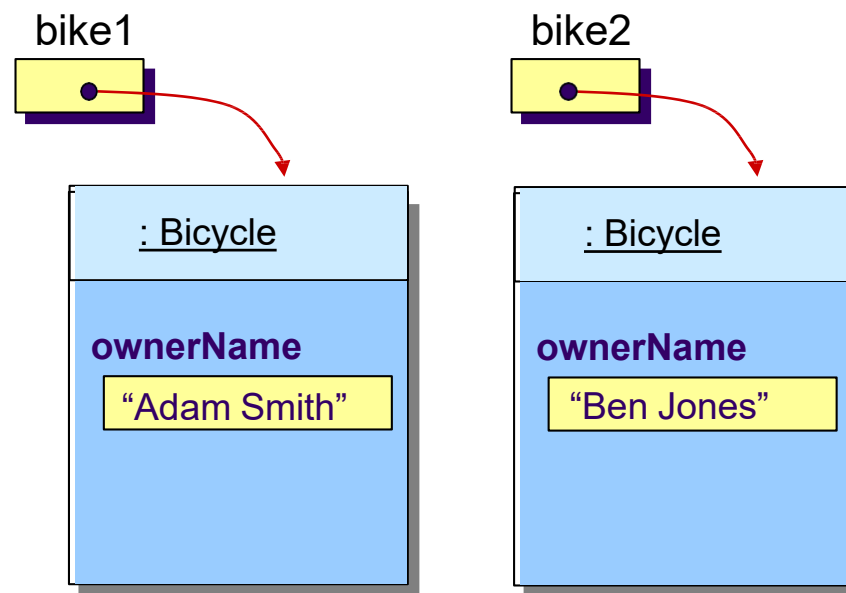
```
class Bicycle {  
    // гишүүн өгөгдөл  
    private String ownerName;  
  
    //Байгуулагч: гишүүн өгөгдлүүдэд утга олгоно  
    public void Bicycle( ) {  
        ownerName = "Unknown";  
    }  
  
    //Дугуйн эзний нэрийг буцаана  
    public String getOwnerName( ) {  
        return ownerName;  
    }  
  
    //дугуйн эзний нэрийг олгоно  
    public void setOwnerName(String name) {  
        ownerName = name;  
    }  
}
```

# Классын инстансууд



- ★ **Bicycle** классыг нэгэнт тодорхойлсон л бол үүнээс хэдэн ч инстанс үүсгэх боломжтой.

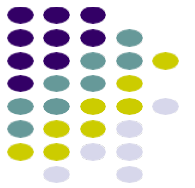
```
Bicycle bike1, bike2;  
bike1 = new Bicycle( );  
bike1.setOwnerName("Adam Smith");  
  
bike2 = new Bicycle( );  
bike2.setOwnerName("Ben Jones");
```



# Асуух үйлдэл (Query)

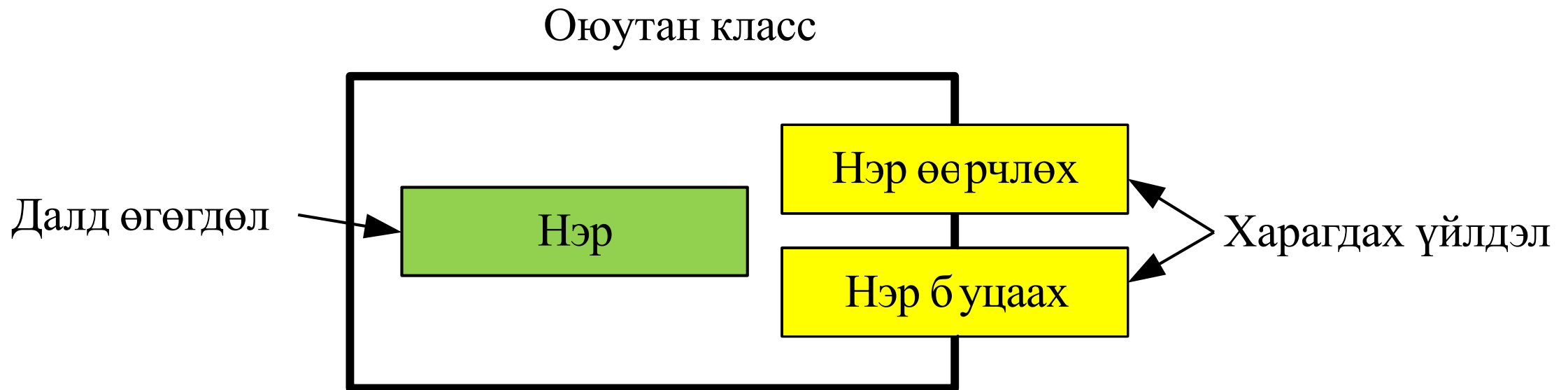


- ★ Зарим үйлдэл "асуух үйлдэл" байдаг.
- ★ Энэ үйлдэл дуудалтын үед ямар нэгэн нөлөө үзүүлэхгүй буюу объектын төлөвт өөрчлөлт орохгүй гэсэн үг.
- ★ Өгөгдлийн утгыг буцаадаг үйлдлийг асуух үйлдэл гэнэ. Мөн getter үйлдэл гэж ярьдаг.



## Өөрчлөх үйлдэл /mutator/

- ★ Хувийн харагдалттай өгөгдлийн утгыг өөрчлөх үйлдлийг өөрчлөх үйлдэл гэнэ. Мөн setter үйлдэл гэдэг.



# Асуух болон өөрчлөх үйлдэл



```
public class Student {  
    private String name;  
  
    public String getName() {  
        return name;  
    }  
    public void setName(String newName) {  
        name = newName;  
    }  
}  
  
...  
Student st1 = new Student();  
st1.setName("bold");  
a = st1.getName();
```





# Нэг үйлдлийн олон хэлбэржилт - overloading

- ★ Нэг классад ижилхэн нэртэй үйлдэл байж болно. Гэхдээ эдгээр нь параметрууд болон буцаах утгаараа ялгаатай байх ёстой. Үүнийг параметрын олон хэлбэржилт гэдэг. Жишээ нь:
  - ★ `Нэмэх(12, 24)`
  - ★ `Нэмэх(3.14, 0.12)`
  - ★ `Нэмэх('a', 'b')`
  - ★ `Нэмэх("hello", "world")`

# Гэр бүлийн гишүүд бүртгэдэг Family класс



```
public class Family {
    private Father father;
    private Mother mother;
    private Daughter[] daughters;
    private int daughterCount=0;
    private int sonCount=0;
    private Son[] sons;
    public Family() {
        daughters = new Daughter[3];
        sons = new Son[3]; }
    public void addMember(Father dad) {
        father = dad; }
    public void addMember(Mother mom) {
        mother = mom; }
    public void addMember(Daughter girl) {
        daughters[daughterCount++] = girl; }
    public void addMember(Son so) {
        sons[sonCount++] = so; }
}
```

```
public class Program {
    public static void main() {
        Family fam = new Family();
        Father f = new Father();
        Mother m = new Mother();
        Daughter d1 = new Daughter();
        Son s1 = new Son();

        fam.addMember(f);
        fam.addMember(m);
        fam.addMember(d1);
        fam.addMember(s1);
    }
}
```

# ДҮГНЭЛТ



- ★ Классын зарлагаа нь гишүүн өгөгдөл, гишүүн үйлдэл, байгуулагч функцыг агуулна.
  - ★ Классыг тодорхойлсон бол үүнээс хэдэн ч объектыг үүсгэж болно.
  - ★ Класс болон объектыг үйлдэл хийлгэхийн тулд мессеж дамжуулах буюу үйлдэл(метод)-ийг дуудна.
  - ★ Үйлдэл
    - ★ Асуух үйлдэл
    - ★ Өөрчлөх үйлдэл
- Эдгээр үйлдлүүд нь классад байх шаардлагатай туслах үйлдлүүд юм.
- Хэрэв классад `private`, `protected` хувьсагч байгаа бол асуух, өөрчлөх үйлдлүүд байна.
  - Эдгээр дээр классын өөрийн араншинг агуулсан үйлдлүүд заавал нэмэгдэх ёстой.