**Гүйцэтгэсэн: Г. Мөнхзаяа/19B1NUM0020/**

МОНГОЛ УЛСЫН ИХ СУРГУУЛЬ

Хэрэглээний Шинжлэх Ухаан Инженерчлэлийн Сургууль

*Док. Д.Энхзол, Маг. Б.Батням, Маг. Ц.Лхамролом*

A1) Жава програмчлалын хэлний дараах түлхүүр үгүүдийг тайлбарлан бичнэ үү.

| | |
|---|---|
| Interface | Interface нь биегүй функцуудын цуглуулга юм. Interface-ээс Implement хийсэн классд method-уудынх нь хэрэгжүүлэлтийг хийдэг. |
| Abstract | Abstract түлхүүр үг нь class болон method-уудад хэрэглэгддэг хандалтын бус хувиргагч юм.<br>**Abstract class:** Объект үүсгэх боломжгүй класс юм. Бусдаар энгийн класстай адил удамшилд ашиглах зорилготой.<br>**Abstract method:** Abstract method-ийг зөвхөн abstract классд ашигладаг бөгөөд биегүй функц байдаг. Удамшсан классд биеийг нь бичих боломжтой. |
| Final | Энэ түлхүүр үгийг ашиглан хувьсагч, функц, классыг тогтмол утгатай болгодог.<br>• Final-аар тодорхойлсон хувьсагчийн утгыг өөрчлөх болохгүй анх өгсөн утга нь л хүчинтэй.<br>• final method-ийг overridden хийж болохгүй.<br>• final классаас удамшуулж болохгүй. |
| Super | Энэ түлхүүр үг нь тухайн классын супер буюу эх классыг илэрхийлнэ. Супер классын аргуудыг дуудах, байгуулагч функц руу нь хандахад хэрэглэгддэг. |

Tut4q.jar файлыг Eclipse дээр импорт хийж, эсвэл задлан доторх классуудыг ашиглан энэ даалгаварыг хийнэ.

A2) Дараах кодыг ажиллуулахад ямар үр дүн өгөх вэ? Үр дүнг ажиглан, хуулбарлан оруулж аль хэсэг онцлог байгааг тайлбарлан бичнэ үү.

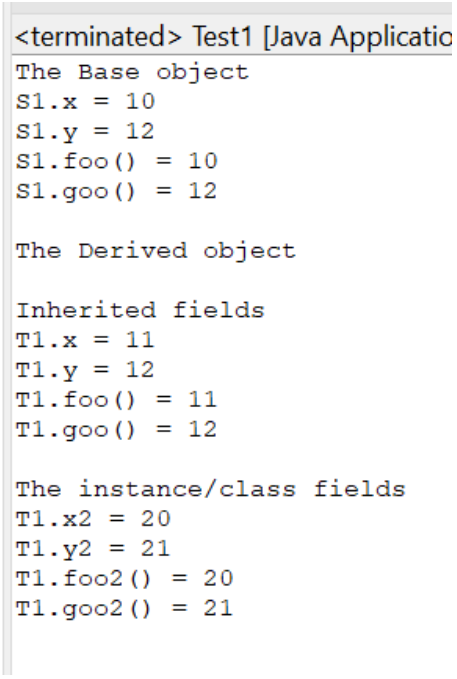
```java
package q1;

public class SuperClass {
	public int x = 10;
	static int y = 10;
	SuperClass() {
		x = y++;
	}
	public int foo() {
		return x;
	}
	public static int goo() {
		return y;
	}
}
```

```java
package q1;

public class Test1 extends SuperClass {
	int x2 = 20;
	static int y2 = 20;

	Test1() {
		x2 = y2++;
	}
	public int foo2() {
		return x2;
	}
	public static int goo2() {
		return y2;
	}
	public static void main(String[] args) {
		// TODO Auto-generated method stub
		SuperClass s1 = new SuperClass();
		Test1 t1 = new Test1();
		System.out.println("The Base object");
		System.out.println("S1.x = " + s1.x);
		System.out.println("S1.y = " + s1.y);
		System.out.println("S1.foo() = " + s1.foo());
		System.out.println("S1.goo() = " + s1.goo());
		System.out.println("\nThe Derived object");
		System.out.println("\nInherited fields");
		System.out.println("T1.x = " + t1.x);
		System.out.println("T1.y = " + t1.y);
		System.out.println("T1.foo() = " + t1.foo());
		System.out.println("T1.goo() = " + t1.goo());
		System.out.println("\nThe instance/class fields");
		System.out.println("T1.x2 = " + t1.x2);
		System.out.println("T1.y2 = " + t1.y2);
		System.out.println("T1.foo2() = " + t1.foo2());
		System.out.println("T1.goo2() = " + t1.goo2());
	}
}
```




Үр дүн:

Base object, class fields:

Супер класс нь s1 болон t1 объектыг үүсгэж байгаа тул 2 удаа дуудагдаж байгаа тул у-ын утга 2-оор нэмэгдэж 12 болсон. Үйлдлийн дарааллын хувьд х нь эхлээд өөртөө у-ын анхныы утга 10-ыг өгөөд нэмэгдүүлж байгаа. Харин у нь статик тул өмнөх нэмэгдсэн утгаа авч байгаа.

Derived object:

T1 нь superclass-аас удамшсан класс тул эх классын байгуулагч 2 удаа дуудагдаж байна. Тиймээс х-ын утга нэгээр нэмэгдсэн. Харин у-ын утга нь base object-ын у-тэй адил.

A3) SuperClass-ыг өөрчлөөгүй бөгөөд Test1 классыг дараахаар өөрчилсөн. Кодыг ажиллуулахад ямар үр дүн өгөх вэ? Үр дүнг ажиглан, хуулбарлан оруулж аль хэсэг онцлог байгааг тайлбарлан бичнэ үү.

```
public class Test1 extends SuperClass {
	static int x = 15;
	static int y = 15;
	int x2 = 20;
	static int y2 = 20;

	Test1() {
		x2 = y2++;
	}
	public int foo2() {
		return x2;
	}
	public static int goo2() {
		return y2;
	}
	public static int goo() {
		return y2;
	}
	public static void main(String[] args) {
		// TODO Auto-generated method stub
		SuperClass s2 = new Test1();
		System.out.println("\nThe static Binding");
		System.out.println("S2.x = " + s2.x);
		System.out.println("S2.y = " + s2.y);
		System.out.println("S2.foo() = " + s2.foo());
		System.out.println("S2.goo() = " + s2.goo());
	}
}
```

Үр дүн:

```
The  static  Binding
S2.x  =  10
S2.y  =  11
S2.foo()  =  10
S2.goo()  =  11
```

S2 нь superclass төрлийн объект учраас superclass-ынхаа x,y-ын утгыг авч байна.

A4) s2.goo() нь аль классаас дуудагдаж буй арга вэ? Яагаад?

SuperClass-аас дуудагдаж байгаа. Учир нь s2 нь SuperClass-ын объект юм.

A5) s2 хувьсагчаас foo2() аргыг дуудахын тулд ямар үйлдэл хийх хэрэгтэй вэ?

(Test1) s2).foo2();

A6) Test1 t2 = new SuperClass(); гэсэн мөрийг ажиллуулахад үр дүн нь юу байх вэ? Яагаад?

Алдаа заана. Учир нь test1 нь superclass-ын дэд/хүүхэд/ класс юм.

A7) Test1 t2 = (Test1) new SuperClass(); гэсэн мөрийг ажиллуулахад үр дүн нь юу байх вэ? Яагаад

Алдаа заана.

A8) Байгуулагч функцийн харагдацийг зааж өгөөгүй байхад үндсэн харагдац нь юу вэ? (default visibility)

Private байна.

A9) Test2.java кодыг ажигла. “The constructor SuperClass() is not visible” алдаа гарч байна. Хэрхэн засах вэ?

Test2-ын байгуулагч функцыг public болгоно.

A10) Interface хэрэгжүүлж байгаа бол түүний хичнээн аргыг хэрэгжүүлэх шаардлагатай вэ?

Interface-д зарлагдсан бүх аргуудыг тодорхойлох ёстой.

A11) Класс X нь интерфейс A-г хэрэгжүүлсэн бөгөөд интерфейс A нь интерфейс B-ээс удамшсан. Тэгвэл A интерфейсд шинэ арга (функц) нэмж өгвөл ямар өөрчлөлт гарах вэ?

A интерфейсд нэмэгдсэн шинэ аргыг X дээр хэрэгжүүлэх хэрэгтэй болно.

A12) Нэг класс хэчнээн интерфейсийг хэрэгжүүлэх боломжтой вэ?

Нэг класс нь хамгийн ихдээ $2^{16}$ -1 interface-ийг хэрэгжүүлж болно.

A13) Хийсвэр класс (abstract class) нь функцыг зарлаад тодорхойлохгүй байж болдгоороо интерфейстэй төстэй.

Гэхдээ зарим функцийн тодорхойлолтыг өөртөө агуулж болдог. Тэгвэл хийсвэр классд шинээр их бие буюу тодорхойлолтгүй функц нэмэх бол хэрхэн тодорхойлох вэ?

Abstract түлхүүр үгийг нэмэх функцынхаа өмнө бичнэ.

A14) Concrete class буюу объект үүсгэж болдог (хийсвэр биш) классд хийсвэр функц тодорхойлж болох уу?

Болохгүй. Зөвхөн abstract класс дотор тодорхойлно.

A15) SubClass1 нь хийсвэр класс A-аас удамшсан болон интерфейс B, C, D-г

хэрэгжүүлсэн бол классын кодын ерөнхий тодорхойлолтыг бичнэ үү. (классын гишүүдийг тодорхойлох шаардлагагүй)

SubClass1 нь B, C, D interface-ийн бүх аргуудыг хэрэгжүүлнэ. Мөн A хийсвэр классын бүх  хийсвэр функцуудыг тодорхойлох ёстой.