Inheritance toreme demekdir ve novleri var

1) Simple Inheritance

2) Multilevel Inheritance

3) Hierarchical (arranged in order of rank) Inheritance

4) Multiple Inheritance

1) Simple Inheritance

Sade toremedir mes. animal classi var, odna catt, dog ve s. torenir

Animal

/ \

Cat Dog

2) Multilevel Inheritance

Animal

/ \

Cat Dog

/

Tiger

Tiger is a Car, and an animal. (Every derived class is base class)

Bu Multilevel Inheritance.

Tiger hem oz, hem Cat-in,hem de animal-in ozelliklerini dasiyir

3) Hierarchical (arranged in order of rank) Inheritance

Hierarchical Inheritance binary treeye benzeyir

Adından da göründüyü kimi, bu, siniflərin iyerarxiyasıdır(siralanmasi). Tək əsas sinif və çoxlu törəmə siniflər var. Bundan əlavə, törəmə siniflər bəzi digər siniflər də tərəfindən törənir Beləliklə, ağaca bənzər bir quruluş iyerarxiya şəklində formalaşır.

Animal

/ \

Cat Dog

/ \ / \

Tiger Scot Pitbull Golden

4) Multiple Inheritance

Bir klass 2 ve daha artiq klassdan torenir

Multiple Inheritance, bir sinif birdən çox əsas sinifdən törəndikdə baş verir. Beləliklə, sinif çoxlu irsiyyətdən istifadə edərək bir neçə əsas sinifdən xüsusiyyətləri miras ala bilərş

Human

/ \

Teacher Student

\ /

Superstudent

Bezi konfliktler bas vere biler

1) mes. 2 base classda eyni adli fieldler var ve onlar derived classda toqqusa biler bunu qarsisini almaq ucun SCOPE RESOLUTIONDAN istifade edirik (BASE1::field, BASE2::field)

2) Diamond problem (solution - "virtual")

Problem:

A A

| |

B C

\ /

D

Burada D ni yaradarken 2 defe A constructoru cagirilir

onun qarsisini almaq ucun B ve C classlarinin qarsisina virtual acar sozu yazilir

Solution with the keyword VIRTUAL

A

/ \

B C

\ /

D

class A

{

public:

A()

{

cout << " Default constructor of A " << endl;

}

};

class B: virtual public A

{

};

class C : virtual public A

{

};

class D : public B, public C

{

};

int main()

{

D d;

return 0;

}

Neticede bir defe A constructoru cagirildi

OUTPUT:

Default constructor of A

Notes:

1)Derived classda bos obyekt yaratmaq ucun base classda bos obyekt yaranmalidir

2)Private fieldler toreme klassda hemise inaccessible (private) olur

3)Torenen klass override edende (base classda olan metodu toreme classda eyni adla yazmaq) artiq base classdaki hemin metodun evezine toreme classdaki hemin override edilmis metod cagirilir. Override-in olmasi ucun virtual acar sozu ve base classa pointer olmalidir