1. Non static members of parent class will get inherited to child class object.

2. The main purpose of inheritance is to reuse the members of parent class

3. In java at class level multiple inheritance is not possible. Multiple inheritance results in complex designing of the software

Example 1:

```
public class A {
	int i = 10;
}

public class B extends A{

	public static void main(String[] args) {
		B b1 = new B();
		System.out.println(b1.i);
	}
}
```

Output

10

Example 2:

```
public class A {
	int i = 10;

	public void test(){
		System.out.println("from test");
	}
```

```
}

public class B extends A{

	public static void main(String[] args) {
		B b1 = new B();
		System.out.println(b1.i);
		b1.test();
	}
}
```

Output:

10

from test

Example 3:

Here A and B are non sub class, as no inheritance is happening between them:

```
public class A {
	int i = 10;

	public void test(){
		System.out.println("from test");
	}
}

public class B {
```

```
	public static void main(String[] args) {
		A a1 = new A();
		System.out.println(a1.i);
		a1.test();
	}
}
```

Output:

10

From Test

Example 4:

```
public class A {

	public void test1(){
		System.out.println("from test");
	}
}
public class B extends A{

	public void test2(){
		System.out.println("From Test 2");
	}
}
public class C extends B{

	public void test3(){
		System.out.println("From test 3");
```

```
	}
	public static void main(String[] args) {
		C c1 = new C();
		c1.test1();
		c1.test2();
		c1.test3();
	}

}
```

Output:

from test

From Test 2

From test 3

Example 5:

```
public class A {

	public void test1(){
		System.out.println("from test");
	}
}
public class B {

	public void test2(){
		System.out.println("From Test 2");
	}
}
```

```
public class C extends A,B{//Error


	public void test3(){
		System.out.println("From test 3");
	}
	public static void main(String[] args) {
	}


}
```

Output:

Error

Example 6:

```
public class A {
	static int i = 10;
}


public class B extends A{


	public static void main(String[] args) {
		B b1 = new B();
		System.out.println(b1.i);//A.i
	}
}
```

Output:

10

Example 7:

```
public class A {
	static int i = 10;
}

public class B extends A{

	public static void main(String[] args) {
		B b1 = new B();
		System.out.println(B.i);//A.i
	}
}
```

Output:

10

Example 8:

```
public class A {
	public static void test(){
		System.out.println("From test");
	}
}

public class B extends A{
```

```
	public static void main(String[] args) {
		B.test(); //A.test()
	}
}
```

Output:

From test