Experiment 3
Source code

Part 1

```
package experiment3t;
import java.util.*;
public class stack_extended extends stack{
	stack_extended(){
		super();
	}
	void push(int a){
		this.a[this.top] = a;
		this.top++;
		this.display();
	}
	 void pop(){
		System.out.println("Message "+a[top-1]);
		this.top--;
		this.display();
	}
	   void display(){
		System.out.println("Current stack contains: ");
		for(int j=0;j<top;j++){
			System.out.println(a[j]+"\t");
		}
	}
}
package experiment3t;
import java.util.*;
public class stack {
	protected int a[];
	protected int top;
	stack(){
		a = new int[5];
		top = 0;
	}
}
package experiment3t;
import java.util.*;
public class Experiment3t {
	public static void main(String[] args) {
		// TODO code application logic herE
		stack_extended es = new stack_extended();
		es.push(5);
		es.push(10);
		es.push(15);
		es.pop();
		es.pop();
		es.pop();
	}
}
```

```
Current stack contains:
5
Current stack contains:
5
10
Current stack contains:
5
10
15
Message 15
Current stack contains:
5
10
Message 10
Current stack contains:
5
Message 5
Current stack contains:
```

Part 2

```
package experiment3_part2;
import java.util.*;
public class Experiment3_part2 {
	public static void main(String[] args) {
		Subdomain s=new Subdomain();
		System.out.println("Single level");
		s.pri();
		Son s1=new Son();
		System.out.println("multilevel");
		s1.pr();
		daughter1 s2=new daughter1();
		System.out.println("heirarchical");
		s2.d1();
		daughter2 s3=new daughter2();
		System.out.println("heirarchical");
		s3.d2();
	}
}
```

Single level

```
package experiment3_part2;
public class Domain {
	void op(){
		System.out.println("This is Domain");
	}
}
```

```
package experiment3_part2;
public class Subdomain extends Domain {
	void pri(){
		super.op();
		System.out.println("This is subdomain");
	}
}
```

Multilevel

```
package experiment3_part2;
public class Grand {
	void m1(){
		System.out.println("This is Grand Father");
	}
}
```

```
package experiment3_part2;
public class Father extends Grand {
	void m2(){
		super.m1();
		System.out.println("This is Father");
	}
}
```

```
package experiment3_part2;
public class Son extends Father {
	void pr(){
		super.m2();
		System.out.println("This is son");
	}
}
```

Hierarchical

```
package experiment3_part2;
public class amoeba {
	void am(){
		System.out.println("This is mother amoeba");
	}
}
```

```
package experiment3_part2;
public class daughter1 extends amoeba {
	void d1(){
		super.am();
		System.out.println("This is first daughter");
	}
}
```

```
package experiment3_part2;
public class daughter2 extends amoeba {
	void d2(){
		super.am();
		System.out.println("This is second daughter");
	}
}
```

```
Single level
This is Domain
This is subdomain
multilevel
This is Grand Father
This is Father
This is son
heirarchical
This is mother amoeba
This is first daughter
heirarchical
This is mother amoeba
This is second daughter
```