Inheritance: Concept Challenge

Concept Challenge

- **Pause** Try to solve the problem yourself
- **Discuss** with other learners (if you can)
- **Watch** the UCSD learners video
- **Confirm** your understanding with our explanation

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

Start IVQ

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

<IVQ placeholder> (B)
Suppose you call:
Student s = new Student();

What is the order of statements printed?

A. #1 #2 #3
B. #1 #3 #2
C. #3 #2 #1
D. #3 #1 #2
E. None of the above

End IVQ / Start Discussion

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
Student s = new Student();
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
Student s = new Student();
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
Student s = new Student();
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
public class Person {
 private String name;
 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

```
public class Person {
 private String name;
 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
Student s = new Student();
```

super?

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
public class Person extends Object {
 private String name;
 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
Student s = new Student();
```

super!

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
Student s = new Student();
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

Output:

#1

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

Output:

#1

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

Output:

#1 #3

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

Output:

#1 #3

```
public class Person {
 private String name;

 public Person( String n ) {
  this.name = n;
  System.out.print("#1 ");
 }
}
```

```
public class Student extends Person {
 public Student () {
  this("Student");
  System.out.print("#2 ");
 }
 public Student( String n ) {
  super(n);
  System.out.print("#3 ");
 }
}
```

```
Student s = new Student();
```

Output:

#1 #3 #2