***Moldovan Rares Valentin Grupa 225*** [***mrir2562@scs.ubbcluj.ro***](mailto:mrir2562@scs.ubbcluj.ro)

**Ce se afiseaza la rularea codului urmator? Explicati.**

**public class** Application {  
  
 **public static void** main(String[] args) {  
 Student student = **new** Student();  
 *updateStudent*(student, **"Name"**, 3);  
 *printUpperCaseStudentName*(student);  
 }  
  
 **private static void** updateStudent(Student student, String newName, **int** newYear) {  
 student = **new** Student(newName, newYear);  
 }  
  
 **private static void** printUpperCaseStudentName(Student student) {  
 System.***out***.println(student.getName().toUpperCase());  
 }  
}  
  
**class** Student {  
  
 **private** String **name**;  
 **private int year**;  
  
 Student() {  
 }  
  
 Student(String name, **int** year) {  
 **this**.**name** = name;  
 **this**.**year** = year;  
 }  
  
 String getName() {  
 **return name**;  
 }  
}

**Variante de raspuns:**

1. null
2. NAME
3. Name
4. Nimic, este aruncata exceptia null pointer exception
5. Eroare de compilare

**Explicatii:**

In Java, parametrii se transmit mereu prin valoare. In consecinta, mereu cand un obiect este transmis ca parametru, se creaza pentru functia apelata o copie a referintei de pe stiva catre acesta. Asta inseamna ca putem modifica in updateStudent orice atribut din obiectul asociat referintei student, insa daca vom asocia o noua referinta, cea din functia main nu se va modica. Campul name al obiectului student din main va ramane null, iar in functia printUpperCaseStudentName, toUpperCase() va arunca Null Pointer Exception.