1. An ATM program is developed to check card type as the ATM will accept payment with  
   Master Cards only would you mention what is the best practice for this piece of code:  
   If(Card.Type == “Premium”)  
   Return False;  
   Else If(Card.Type == “Master”)  
   Accept Transaction;  
   Else if(Card.Type == “Youth”)  
   Return False;  
   Else  
   Return Error;

**Answer:**

* **Use explicit boolean values instead of strings to represent the validity of the card. For example, instead of returning "False", the code should return "false".**
* **Use an "if-else if-else" statement instead of multiple "if" statements. This makes the code more readable and efficient.**
* **Use a switch statement instead of if-else if-else statements in case there are more card types to check.**

switch (card.getType()) {

case "Premium":

case "Youth":

return false;

case "Master":

acceptTransaction();

break;

default:

return error;

}

1. Find the error in the following piece of code:  
   int n = 0;  
   While( n<10)  
   {  
   int a =n+2;

int b =a+n;  
 System.out.println(“a = ”+a);  
 n=5;  
 }  
 System.out.println(“a = “+a);  
 System.out.println(“b = “+b);  
 System.out.println(“n = “+n);

1. There are a few errors in the given piece of code:

* The "w" in "while" should be lowercase.
* The initialization of the variable "a" and "b" should be outside of the while loop.
* The variable "a" and "b" are not in scope outside of the while loop, so the print statements at the end will not compile.

int a = 0;

int n = 0;

int b = 0;

while (n < 10) {

a = n + 2;

b = a + n;

System.out.println("a = " + a);

n = 5;

}

System.out.println("a = " + a);

System.out.println("b = " + b);

System.out.println("n = " + n);