

Question 3:

Code 1:

```
int n = 3;
int m = 4;
int result = n * (++m);
```

Explanation1:

The output of this program would be 15, as it would be $3 * 5$. This is due to the fact that “++m” means that m is incremented before executing, meaning $m = 5$ before finishing the calculation.

Code 2:

```
int n = 3;
int m = 4;
int result = n * (m++);
```

Explanation2:

The output would be 12, as m in this case would still be 4. It is only after the calculation is done would m be incremented.

A:

What I noticed is that compound interest is an exponential growth over the years vs simple interest which is just a linear growth. As a result, the difference between both types of interest also becomes exponential.

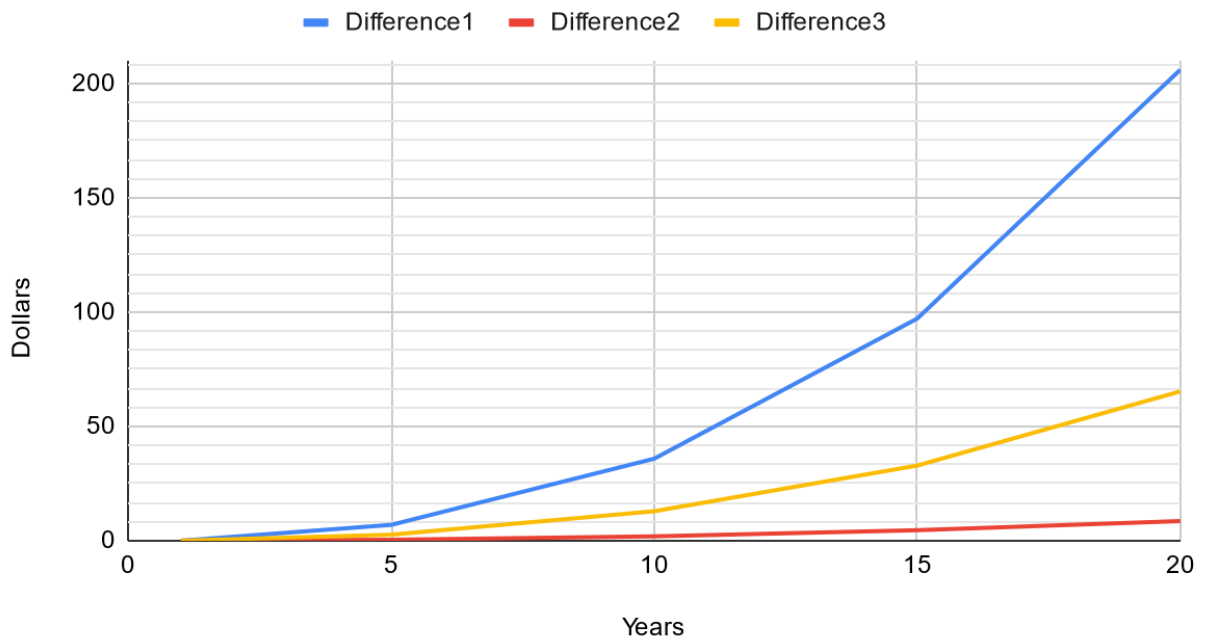
B:

I managed to notice that for each percentage increase, the difference is still exponential, as it scales the higher the percentage is. I chose 2% and 8% and it can be seen in the 2% interest that there is still a slight increase that can be compared to an exponential curve. With the 8% it is even more pronounced.

C: Based on my implementation as well as research, you theoretically would be able to encode the interest as a “final” variable, as the variable itself does not need to change through output the execution of the program. However, since a “static” just means that it is public to all classes but also a static variable can not be declared in another method. In my program, you will not be able to use static because of the fact that the variable interest gets its input from the main method through arguments, which throws an error that “modifier static is not allowed here”. However, someone could use the scanner function after importing “scanner” to get input of the interest BEFORE the executing “public static void main(String[] args)” part of the program, which would then allow the static variable.

Graphs:

Differences Vs Interest Rate



Simple, Compound and Difference

