

FUNCTION OUTPUT

```
pv <- function(FV, r, n) {  
  present_value <- FV / (1 + r)^n  
  round(present_value, 2)  
}
```

```
pv(FV = 1000, r = .08, n = 5)  
[1] 680.58
```

```
pv2 <- function(FV, r, n) {  
  present_value <- FV / (1 + r)^n  
  return(present_value)  
  round(present_value, 2)  
}
```

```
pv2(1000, .08, 5)  
[1] 680.5832
```

What gets returned from a function is either:

1. The last expression evaluated
2. `return(value)`, which forces the function to stop execution and return value

FUNCTION OUTPUT

```
pv <- function(FV, r, n) {  
  present_value <- FV / (1 + r)^n  
  round(present_value, 2)  
}
```

```
pv(FV = 1000, r = .08, n = 5)  
[1] 680.58
```

```
pv2 <- function(FV, r, n) {  
  present_value <- FV / (1 + r)^n  
  return(present_value)  
  round(present_value, 2)  
}
```

```
pv2(1000, .08, 5)  
[1] 680.5832
```

What gets returned from a function is either:

1. The last expression evaluated
2. `return(value)`, which forces the function to stop execution and return value

Note the differences in how we call these functions. Why do both cases work?