**Example of Functions and Arguments**

|  |
| --- |
| ### A simple mortgage calculator  calcMortgage <- function( years, APR, principal )  {  months <- years \* 12  int.rate <- APR / 12  monthly.payment <- ( principal \* int.rate ) /  (1 - (1 + int.rate)^(-months) )  return( monthly.payment )    }  # Calculate monthly payments for a 30-year mortgage #  # on a $100,000 loan at a 5% APR interest rate: #  calcMortgage(  years=30,  APR=0.05,  principal=100000  )  [1] 536.8216 # output from the console  ### Does not run:  calcMortgage( principal=250000 )  # here the function does not know the term of the loan or rate  ### ‘throw’ plus ‘catch’ in order to save the value  my.payment <- calcMortgage( years=30, APR=0.05, principal=100000 )  >my.payment  [1] 536.8216  ### Example of a function with some default values provided  calcMortgage <- function( years=30, APR=0.05, principal )  {  months <- years \* 12  int.rate <- APR / 12  monthly.payment <- ( principal \* int.rate ) /  (1 - (1 + int.rate)^(-months) )  return( monthly.payment )    }  # the function now runs, because if you don’t provide it with  # an argument it assumes you want to use the default values.  calcMortgage( principal=250000 )  [1] 1342.054  # you can over-write default values by supplying an argument  calcMortgage( years=15, principal=250000 )  [1] 1976.984 |