Use Case 3.1	**Create ConstantRateConstant**
Primary Actor:	End-User
Preconditions:	• None
Postconditions:	• The ConstantRateConstant is initialized and valid

Main Success Scenario:

1. The user calls ConstantRateConstant's parameterized constructor with its name, and rate, k, as the two arguments
2. The ConstantRateConstant is initialized successfully

Extensions:

2a. The ConstantRateConstant's name is an empty string

1. System throws a runtime exception notifying the user that the name cannot be empty

Use Case 3.2	**Get ConstantRateConstant Rate**
Primary Actor:	End-User
Preconditions:	• The ConstantRateConstant is initialized and valid
Postconditions:	• The rate is returned and the ConstantRateConstant is unmodified

Main Success Scenario:

1. The user calls ConstantRateConstant's getRate() function with a smart_ptr to a StateOfTheWorld instance as the only argument (StateOfTheWorld does not have to be valid)
2. The rate is calculated as (k * 1) and is returned to the user

Extensions:

1. None