CH11_005

ANSWER

ADDING CONSOLE FUNCTIONS AND METHODS

1 mySimObject5

floatingSums() - Finished Code:

```
ConsoleMethod( mySimObject5 , floatingSums , F32, 2, 0,
              "obj.floatingSums( [ float0 [, float1 [, ... ] ] ])"
              " - Takes 0 or more floating point numbers, adds them,"
              " and then returns the sum." )
{
   int numFloats = argc - 2;

   F32 sumTotal = 0.0f;

   for( int count = 0; count < numFloats; count++ )
   {
      sumTotal += dAtof( argv[2+count] );
   }

   return sumTotal;
}
```

ADDING CONSOLE FUNCTIONS AND METHODS

tokenize() - Finished Code:

```
ConsoleFunction( tokenize, const char* , 4 , 0 ,
                "tokenize( token, string0, string1 [ , ... ] )"
                " - Returns combined set of strings, separated by"
                " token." )
{
   // Ignore method name, object ID, and first two required args
   int numOptionalStrings = argc - 4;
   const char *token = argv[1];
   int tokenLen = dStrlen( token );
   int bufferLen = tokenLen + dStrlen( argv[2] ) + dStrlen( argv[3] );

   // Count total length of string
   for( int count = 0; count < numOptionalStrings; count++ )
   {
      bufferLen += ( tokenLen + dStrlen( argv[4 + count] ) );
   }

   bufferLen++; // Add one space for closing NULL

   // Allocate and initialize the return bufffer
   char *returnBuffer = Con::getReturnBuffer( bufferLen );
   dMemset( returnBuffer , '\0' , bufferLen );

   // Fill the buffer
   dSprintf( returnBuffer, bufferLen, "%s%s%s", argv[2], token, argv[3] );

   int tmpLen = dStrlen( returnBuffer );

   for( int count = 0; count < numOptionalStrings; count++ )
   {
      dSprintf( returnBuffer + tmpLen, bufferLen - tmpLen, "%s%s",
                token, argv[4 + count] );

      tmpLen += tokenLen + dStrlen( argv[4 + count] );
   }
   return returnBuffer;
}
```