

# Package ‘GenePackage’

August 15, 2024

**Type** Package

**Title** A Package for Gene Objects

**Description** This package provides S4 classes and methods for representing different types of genes, including protein-coding genes, long non-coding RNAs, microRNAs, and transcripts. It includes functionalities for accessing gene attributes, calculating gene product lengths, and determining GC content.

**Version** 0.1.0

**Author** Negin Kafee [aut, cre]

**Maintainer** Negin Kafee <negin.kafee@yahoo.com>

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**biocViews** Genomics, Transcriptomics, GeneExpression, GeneRegulation,  
GenomeAnnotation

**Imports** methods

**Suggests** rmarkdown, testthat (>= 3.0.0)

**URL** <https://github.com/negin-kafee/GenePackage>

**BugReports** <https://github.com/negin-kafee/GenePackage/issues>

**NeedsCompilation** no

## Contents

createGene . . . . .	2
gcContent . . . . .	2
Gene-class . . . . .	3
getSymbol . . . . .	3
lengthProduct . . . . .	4
LongNonCodingRNA-class . . . . .	4
MicroRNA-class . . . . .	5
ProteinCodingGene-class . . . . .	5
setChromosome . . . . .	5
setDescription . . . . .	6
setEnd . . . . .	6
setName . . . . .	7

setStart . . . . .	7
setStrand . . . . .	8
setSymbol . . . . .	8
Transcript-class . . . . .	9

<b>Index</b>	<b>10</b>
--------------	-----------

---

createGene	<i>Create a Gene Object</i>
------------	-----------------------------

---

## Description

This function creates a gene object of the specified type.

## Usage

```
createGene(type, ...)
```

## Arguments

type	A character string specifying the type of gene ('protein_coding', 'lncRNA', 'miRNA', 'transcript').
...	Additional parameters for the gene object.

## Value

An object of the specified gene class.

## Examples

```
gene <- createGene("protein_coding", ID = "ENSG000001", symbol = "TP53",
  name = "Tumor protein p53", description = "Plays a role in cell cycle regulation",
  chromosome = "17", start = 7565097, end = 7590856, strand = "+",
  proteinID = "NP_000537", proteinSequence = "MDM2", exonCount = 11L)
getSymbol(gene)
lengthProduct(gene)
gcContent(gene)
```

---

gcContent	<i>Calculate the GC Content of a Gene Sequence</i>
-----------	----------------------------------------------------

---

## Description

This method calculates the GC content (percentage of G and C nucleotides) of a gene's sequence.

## Usage

```
gcContent(object)
```

## Arguments

object	An object of class ProteinCodingGene or Transcript.
--------	-----------------------------------------------------

**Value**

The GC content as a percentage.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", symbol = "TP53",
  proteinSequence = "MDM2")
gcContent(gene)
transcript <- createGene("transcript", ID = "ENST000002", sequence = "ATGCATGC")
gcContent(transcript)
```

---

Gene-class	<i>Gene Class</i>
------------	-------------------

---

**Description**

A class representing a general gene structure with various attributes.

**Slots**

ID A character string representing the gene ID.  
 symbol A character string representing the gene symbol.  
 name A character string representing the gene name.  
 description A character string representing the gene description.  
 chromosome A character string representing the chromosome.  
 start A numeric value representing the start position of the gene.  
 end A numeric value representing the end position of the gene.  
 strand A character string representing the DNA strand.

---

getSymbol	<i>Get the Symbol of a Gene Object</i>
-----------	----------------------------------------

---

**Description**

This method returns the symbol of a gene object.

**Usage**

```
## S4 method for signature 'Gene'
getSymbol(object)
```

**Arguments**

object An object of class Gene.

**Value**

The symbol of the gene.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", symbol = "TP53")
getSymbol(gene)
```

---

lengthProduct	<i>Get the Length of the Gene Product</i>
---------------	-------------------------------------------

---

**Description**

This method returns the length of the gene product, depending on the gene type.

**Usage**

```
lengthProduct(object)
```

**Arguments**

object            An object of a specific gene class.

**Value**

The length of the gene product.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", symbol = "TP53",
  proteinSequence = "MDM2")
lengthProduct(gene)
```

---

LongNonCodingRNA-class	<i>Long Non-Coding RNA Gene Class</i>
------------------------	---------------------------------------

---

**Description**

A class representing a long non-coding RNA gene with specific attributes.

**Slots**

lncRNAID    A character string representing the lncRNA ID.

RNASequence    A character string representing the RNA sequence.

---

MicroRNA-class	<i>MicroRNA Gene Class</i>
----------------	----------------------------

---

**Description**

A class representing a microRNA gene with specific attributes.

**Slots**

miRNAID A character string representing the miRNA ID.

seedSequence A character string representing the miRNA seed sequence.

---

ProteinCodingGene-class	<i>Protein-Coding Gene Class</i>
-------------------------	----------------------------------

---

**Description**

A class representing a protein-coding gene with specific attributes.

**Slots**

proteinID A character string representing the protein ID.

proteinSequence A character string representing the protein sequence.

exonCount An integer representing the number of exons.

---

setChromosome	<i>Set the Chromosome of a Gene Object</i>
---------------	--------------------------------------------

---

**Description**

This method sets the chromosome of a gene object.

**Usage**

```
setChromosome(object, value)
```

**Arguments**

object An object of class Gene.

value The new chromosome value to assign.

**Value**

The modified gene object.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", chromosome = "17")
gene <- setChromosome(gene, "18")
```

---

setDescription	<i>Set the Description of a Gene Object</i>
----------------	---------------------------------------------

---

**Description**

This method sets the description of a gene object.

**Usage**

```
setDescription(object, value)
```

**Arguments**

object	An object of class Gene.
value	The new description value to assign.

**Value**

The modified gene object.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", description = "Tumor protein p53")
gene <- setDescription(gene, "New Description")
```

---

setEnd	<i>Set the End Position of a Gene Object</i>
--------	----------------------------------------------

---

**Description**

This method sets the end position of a gene object.

**Usage**

```
setEnd(object, value)
```

**Arguments**

object	An object of class Gene.
value	The new end position value to assign.

**Value**

The modified gene object.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", end = 7590856)
gene <- setEnd(gene, 7591000)
```

---

`setName`*Set the Name of a Gene Object*

---

**Description**

This method sets the name of a gene object.

**Usage**

```
setName(object, value)
```

**Arguments**

<code>object</code>	An object of class <code>Gene</code> .
<code>value</code>	The new name value to assign.

**Value**

The modified gene object.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", name = "Tumor protein p53")
gene <- setName(gene, "New Gene Name")
```

---

`setStart`*Set the Start Position of a Gene Object*

---

**Description**

This method sets the start position of a gene object.

**Usage**

```
setStart(object, value)
```

**Arguments**

<code>object</code>	An object of class <code>Gene</code> .
<code>value</code>	The new start position value to assign.

**Value**

The modified gene object.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", start = 7565097)
gene <- setStart(gene, 7566000)
```

---

setStrand	<i>Set the Strand of a Gene Object</i>
-----------	----------------------------------------

---

**Description**

This method sets the strand of a gene object.

**Usage**

```
setStrand(object, value)
```

**Arguments**

object	An object of class Gene.
value	The new strand value to assign.

**Value**

The modified gene object.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", strand = "+")
gene <- setStrand(gene, "-")
```

---

setSymbol	<i>Set the Symbol of a Gene Object</i>
-----------	----------------------------------------

---

**Description**

This method sets the symbol of a gene object.

**Usage**

```
setSymbol(object, value)
```

**Arguments**

object	An object of class Gene.
value	The new symbol value to assign.

**Value**

The modified gene object.

**Examples**

```
gene <- createGene("protein_coding", ID = "ENSG000001", symbol = "TP53")
gene <- setSymbol(gene, "NEW_SYMBOL")
```



---

Transcript-class	<i>Transcript Class</i>
------------------	-------------------------

---

**Description**

A class representing a transcript, inheriting from the Gene class.

**Slots**

transcriptID A character string representing the transcript ID.

sequence A character string representing the transcript sequence.

# Index

`createGene`, [2](#)

`gcContent`, [2](#)  
`gcContent`, `ProteinCodingGene`-method  
    (`gcContent`), [2](#)  
`gcContent`, `Transcript`-method  
    (`gcContent`), [2](#)

`Gene`-class, [3](#)  
`getSymbol`, [3](#)  
`getSymbol`, `Gene`-method (`getSymbol`), [3](#)

`lengthProduct`, [4](#)  
`lengthProduct`, `LongNonCodingRNA`-method  
    (`lengthProduct`), [4](#)  
`lengthProduct`, `MicroRNA`-method  
    (`lengthProduct`), [4](#)  
`lengthProduct`, `ProteinCodingGene`-method  
    (`lengthProduct`), [4](#)  
`lengthProduct`, `Transcript`-method  
    (`lengthProduct`), [4](#)  
`LongNonCodingRNA`-class, [4](#)

`MicroRNA`-class, [5](#)

`ProteinCodingGene`-class, [5](#)

`setChromosome`, [5](#)  
`setChromosome`, `Gene`-method  
    (`setChromosome`), [5](#)  
`setDescription`, [6](#)  
`setDescription`, `Gene`-method  
    (`setDescription`), [6](#)  
`setEnd`, [6](#)  
`setEnd`, `Gene`-method (`setEnd`), [6](#)  
`setName`, [7](#)  
`setName`, `Gene`-method (`setName`), [7](#)  
`setStart`, [7](#)  
`setStart`, `Gene`-method (`setStart`), [7](#)  
`setStrand`, [8](#)  
`setStrand`, `Gene`-method (`setStrand`), [8](#)  
`setSymbol`, [8](#)  
`setSymbol`, `Gene`-method (`setSymbol`), [8](#)

`Transcript`-class, [9](#)