

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Contents

1.0 OVERVIEW

1.1 INTRODUCTION

1.2 STATISTICS

1.3 FEATURES

1.4 BASIC USAGE

2.0 CLASSES

2.1 CNG.CHARACTER

2.2 CNG.GENDER

2.3 CNG.ORIGIN

2.4 CNG.NAMEGENERATOR

2.5 CNG.SETTINGS

2.6 CNG.NAMEDATA

2.7 CNG.UNICODEMANAGER

2.8 CNG.JSONMANAGER

3.0 UNICODE

3.1 UNICODE, ASCII, AND SPECIAL CHARACTERS

3.2 CONVERSION TABLE

4.0 EXPANDING AND MODIFYING

4.1 ADDING GIVEN OR FAMILY NAMES

4.2 ADDING NAME PREFIXES OR SUFFIXES

4.3 ADDING NICKNAMES

4.4 ADDING GENDERS AND PRONOUNS

4.5 ADDING NAME FORMATS

1.0 OVERVIEW

1.1 INTRODUCTION

Character Name Generator is a Unity Plug-in designed to easily generate random character names, nicknames, genders, and origins while remaining robust, modifiable, and expandable.

1.2 STATISTICS

- Name Prefixes : 101
 - Male : 64
 - Female : 73
 - Unisex : 36
- Given Names : 61,896
 - Male : 43,020
 - Female : 41,829
 - Unisex : 22,953
- Family Names : 158,024
- Name Suffixes : 10
- Nicknames : 1,515
 - Male : 1,429
 - Female : 1,365
 - Unisex : 1,279
- Origins : 101
 - Categories : 7
 - Subcategories : 22
- Genders : 3
- Pronouns : 3
- Total Possible Name Combinations : 91,864,015,038,900,000,000,000

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

1.3 FEATURES

The Character Name Generator includes several features to allow for flexibility.

- No setup required! Works straight out of the box
- Generates name prefixes, given names, family names, name suffixes, and nicknames
- Generates genders, preferred pronouns, and origins
- Filter outcomes by gender, name format, origin name, origin subcategory, and/or origin category
- Prevents reusing recently generated names
- Easily modifiable generator settings
- Toggle support for Unicode characters
- Loads data from easily readable JSON files
- Includes JSON wrapper class

1.4 BASIC USAGE

Implementing the Character Name Generator into your project can be done in three simple steps:

- Include the CNG namespace
 - `using CNG;`
- Load the name data at startup or whenever loading would be most optimal
 - `NameData.LoadNameData ();`
- Generate a character
 - `Character myCharacter = NameGenerator.GetCharacter ();`

2.0 CLASSES

2.1 CNG.CHARACTER

Description: A class for storing data for a character.

Variables

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **Prefix**
Type: string
Description: The name prefix for the character.
- **GivenName**
Type: string
Description: The given name for the character. If the character has multiple given names, the names will be listed one after the other (e.g. Ashley Marie).
- **FamilyName**
Type: string
Description: The family name for the character. If the character has multiple family names, the names will be listed one after the other (e.g. Johnson Green).
- **Suffix**
Type: string
Description: The name suffix for the character.
- **Nickname**
Type: string
Description: The nickname for the character.
- **Gender**
Type: Gender.Label
Description: The gender of the character.
- **Pronouns**
Type: Gender.Pronouns
Description: The preferred pronouns for the character.
- **Origin**
Type: Origin
Description: The origin for the character. All names for this characters will be assigned based on this origin to some degree.

Properties

- **WesternNameOrder**
Type: string
Description: The character's full name constructed in the Western naming order (e.g. Prefix + Given + Family + Suffix).
- **EasternNameOrder**
Type: string
Description: The character's full name constructed in the Eastern naming order (e.g. Prefix + Family + Given + Suffix).
- **QuotedNickname**
Type: string

Description: The character's nickname placed in quotes (e.g. "Tiger").

1.2 CNG.GENDER

Description: A class for handling data associated with a character's gender.

Enums

- **Label**

Description: A typed set of gender labels

Values:

- NONE
- MALE
- FEMALE
- NON_BINARY

- **Pronouns**

Description: A typed set of pronouns to be associated with a gender(s).

Values:

- NONE
- HE_HIM_HIS
- SHE_HER_HERS
- THEY_THEM_THEIR

Functions

- **ToString (Label gender, bool isAbbreviated = false)**

Return Type: string

Description: Get a string of the gender label.

Parameters:

- **gender**
Type: Label
Description: The gender label to be converted into a string.
- **isAbbreviated**
Type: bool
Default Value: false
Description: Whether or not the gender label should be abbreviated.

- **GetSubjectivePronoun (Pronouns pronoun, bool isCapitalized = true)**

Return Type: string

Description: Get the third-person subjective pronoun (e.g. he, she, they).

Parameters:

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **pronoun**
Type: Pronouns
Description: The associated set of pronouns.
- **isCapitalized**
Type: bool
Default Value: true
Description: Whether or not the string returned should be capitalized.
- **GetObjectivePronoun (Pronouns pronoun, bool isCapitalized = true)**
Return Type: string
Description: Get the third-person objective pronoun (e.g. him, her, them).
Parameters:
 - **pronoun**
Type: Pronouns
Description: The associated set of pronouns.
 - **isCapitalized**
Type: bool
Default Value: true
Description: Whether or not the string returned should be capitalized.
- **GetPossessivePronoun (Pronouns pronoun, bool isCapitalized = true)**
Return Type: string
Description: Get the third-person possessive pronoun (e.g. his, hers, their).
Parameters:
 - **pronoun**
Type: Pronouns
Description: The associated set of pronouns.
 - **isCapitalized**
Type: bool
Default Value: true
Description: Whether or not the string returned should be capitalized.

2.2 CNG.ORIGIN

Description: A class for storing origin data for a name.

Enums

- **CategoryType**
Description: The types of group for an origin.
Values:
 - NONE
 - AFRICAN

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- AMERICAN
- ASIAN
- EUROPEAN
- INDIGENOUS
- OTHER

- **SubcategoryType**

Description: The types of subgroup for an origin.

Values:

- NONE
- NORTHERN_AFRICAN
- SOUTHERN_AFRICAN
- WESTERN_AFRICAN
- EASTERN_AFRICAN
- CENTRAL_AMERICAN
- NORTHERN_AMERICAN
- SOUTHERN_AMERICAN
- CENTRAL_ASIAN
- SOUTHERN_ASIAN
- SOUTHEASTERN_ASIAN
- WESTERN_ASIAN
- EASTERN_ASIAN
- NORTHERN_EUROPEAN
- SOUTHERN_EUROPEAN
- WESTERN_EUROPEAN
- EASTERN_EUROPEAN
- OCEANIC
- AMERICAN
- FANTASY
- SCIENCE_FICTION
- OTHER

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Variables

- **Name**
Type: string
Default Value: "None"
Description: The name of the origin
- **Subcategory**
Type: Subcategory
Default Value: Subcategory.NONE
Description: The type of the origin's subcategory.
- **Category**
Type: Category
Default Value: Category.NONE
Description: The type of the origin's category.

Functions

- **ToString (SubcategoryType subcategory)**
Return Type: string
Description: Converts a subcategory to a readable string.
Parameters:
 - **subcategory**
Type: SubcategoryType
Description: The subcategory to be converted.
- **ToString (CategoryType category)**
Return Type: string
Description: Converts a category to a readable string.
Parameters:
 - **category**
Type: CategoryType
Description: The category to be converted.
- **SubcategoryFromString (string subcategory)**
Return Type: SubcategoryType
Description: Converts a string to a subcategory.
Parameters:
 - **subcategory**
Type: string
Description: The string to be converted.
- **CategoryFromString (string category)**
Return Type: CategoryType
Description: Converts a string to a category.
Parameters:

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **category**
Type: CategoryType
Description: The string to be converted.

2.3 CNG.NAMEGENERATOR

Description: A class for generating random characters, genders, origins, formats, prefixes, given names, family names, suffixes, and nicknames.

Enums

- **Format**
Description: The types of name structures for a character.
Values:
 - NONE
 - GIVEN_FAMILY
 - GIVEN_GIVEN_FAMILY
 - GIVEN_FAMILY_FAMILY
 - GIVEN_GIVEN_FAMILY_FAMILY

Functions

- **GetCharacter ()**
Return Type: Character
Description: Generates a random character.
- **GetCharacter (string origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: Character
Description: Generates a random character based on a specific origin name.
Parameters:
 - **origin**
Type: string
Description: The origin name the character should be based on. Passing "", "None", or "Any" will generate a character with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the character should be based on. Passing Gender.Label.NONE will generate a character with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Description: The name format the character should match. Passing Format.NONE will generate a character to match a random format.

- **GetCharacter (Origin.SubcategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: Character

Description: Generates a random character based on a specific origin subcategory.

Parameters:

- **origin**

Type: Origin.SubcategoryType

Description: The origin subcategory the character should be based on. Passing Origin.SubcategoryType.NONE will generate a character with a random origin.

- **gender**

Type: Gender.Label

Default Value: Gender.Label.NONE

Description: The gender the character should be based on. Passing Gender.Label.NONE will generate a character with a random gender.

- **format**

Type: Format

Default Value: Format.NONE

Description: The name format the character should match. Passing Format.NONE will generate a character to match a random format.

- **GetCharacter (Origin.CategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: Character

Description: Generates a random character based on a specific origin category.

Parameters:

- **origin**

Type: Origin.CategoryType

Description: The origin category the character should be based on. Passing Origin.CategoryType.NONE will generate a character with a random origin.

- **gender**

Type: Gender.Label

Default Value: Gender.Label.NONE

Description: The gender the character should be based on. Passing Gender.Label.NONE will generate a character with a random gender.

- **format**

Type: Format

Default Value: Format.NONE

Description: The name format the character should match. Passing Format.NONE will generate a character to match a random format.

- **GetCharacter (Origin origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: Character
Description: Generates a random character based on a specific origin.
Parameters:
 - **origin**
Type: Origin
Description: The origin the character should be based on. Passing an invalid origin will generate a character with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the character should be based on. Passing Gender.Label.NONE will generate a character with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE
Description: The name format the character should match. Passing Format.NONE will generate a character to match a random format.
- **GetGender ()**
Return Type: Gender.Label
Description: Randomly generates a gender based on the gender settings.
- **GetPronouns (Gender.Label gender)**
Return Type: Gender.Pronouns
Description: Assigns a set of pronouns for a gender.
Parameters:
 - **gender**
Type: Gender.Label
Description: The gender to assign pronouns to.
- **GetOrigin (string origin)**
Return Type: Origin
Description: Generates an origin from any or an existing origin name.
Parameters:
 - **origin**
Type: string
Description: The name of the origin. Passing "", "None", or "Any" will generate a random origin.
- **GetOrigin (Origin.SubcategoryType subcategory)**
Return Type: Origin
Description: Generates an origin from any or an existing origin subcategory.
Parameters:

- **origin**
Type: Origin.SubcategoryType
Description: The origin subcategory. Passing Origin.SubcategoryType.NONE will generate a random origin.
- **GetOrigin (Origin.CategoryType category)**
Return Type: Origin
Description: Generates an origin from any or an existing origin category.
Parameters:
 - **origin**
Type: Origin.CategoryType
Description: The origin category. Passing Origin.CategoryType.NONE will generate a random origin.
- **GetFormat ()**
Return Type: Format
Description: Generates a random name format based on the Double Name Chance Settings.
- **GetPrefix ()**
Return Type: string
Description: Generates a random name prefix.
- **GetPrefix (string origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random name prefix based on a specific origin name.
Parameters:
 - **origin**
Type: string
Description: The origin name the prefix should be based on. Passing "", "None", or "Any" will generate a prefix with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the prefix should be based on. Passing Gender.Label.NONE will generate a prefix with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE
Description: The name format the prefix should match. Passing Format.NONE will generate a prefix to match a random format.
- **GetPrefix (Origin.SubcategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Description: Generates a random name prefix based on a specific origin subcategory.

Parameters:

- **origin**
Type: Origin.SubcategoryType
Description: The origin subcategory the prefix should be based on. Passing Origin.SubcategoryType.NONE will generate a prefix with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the prefix should be based on. Passing Gender.Label.NONE will generate a prefix with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the prefix should match. Passing Format.NONE will generate a prefix to match a random format.

- **GetPrefix (Origin.CategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Description: Generates a random name prefix based on a specific origin category.

Parameters:

- **origin**
Type: Origin.CategoryType
Description: The origin category the prefix should be based on. Passing Origin.CategoryType.NONE will generate a prefix with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the prefix should be based on. Passing Gender.Label.NONE will generate a prefix with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the prefix should match. Passing Format.NONE will generate a prefix to match a random format.

- **GetPrefix (Origin origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Description: Generates a random name prefix based on a specific origin.

Parameters:

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **origin**
Type: Origin
Description: The origin the prefix should be based on. Passing an invalid origin will generate a prefix with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the prefix should be based on. Passing Gender.Label.NONE will generate a character with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the prefix should match. Passing Format.NONE will generate a prefix to match a random format.
- **GetGivenName ()**
Return Type: string
Description: Generates a random given name.
- **GetGivenName (string origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random given name based on a specific origin name.
Parameters:
 - **origin**
Type: string
Description: The origin name the given name should be based on. Passing "", "None", or "Any" will generate a given name with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the given name should be based on. Passing Gender.Label.NONE will generate a given name with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE
Description: The name format the given name should match. Passing Format.NONE will generate a given name to match a random format.
- **GetGivenName (Origin.SubcategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random given name based on a specific origin subcategory.
Parameters:

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **origin**
Type: Origin.SubcategoryType
Description: The origin subcategory the given name should be based on. Passing Origin.SubcategoryType.NONE will generate a given name with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the given name should be based on. Passing Gender.Label.NONE will generate a given name with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the given name should match. Passing Format.NONE will generate a given name to match a random format.
- **GetGivenName (Origin.CategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random given name based on a specific origin category.
Parameters:
 - **origin**
Type: Origin.CategoryType
Description: The origin category the given name should be based on. Passing Origin.CategoryType.NONE will generate a given name with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the given name should be based on. Passing Gender.Label.NONE will generate a given name with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE
Description: The name format the given name should match. Passing Format.NONE will generate a given name to match a random format.
- **GetGivenName (Origin origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random given name based on a specific origin.
Parameters:

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **origin**
Type: Origin
Description: The origin the given name should be based on. Passing an invalid origin will generate a given name with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the given name should be based on. Passing Gender.Label.NONE will generate a given name with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the given name should match. Passing Format.NONE will generate a given name to match a random format.
- **GetFamilyName ()**
Return Type: string
Description: Generates a random family name.
- **GetFamilyName (string origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random family name based on a specific origin name.
Parameters:
 - **origin**
Type: string
Description: The origin name the family name should be based on. Passing "", "None", or "Any" will generate a family name with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the family name should be based on. Passing Gender.Label.NONE will generate a family name with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE
Description: The name format the family name should match. Passing Format.NONE will generate a family name to match a random format.
- **GetFamilyName (Origin.SubcategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random family name based on a specific origin

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

subcategory.

Parameters:

- **origin**

Type: Origin.SubcategoryType

Description: The origin subcategory the family name should be based on. Passing Origin.SubcategoryType.NONE will generate a family name with a random origin.

- **gender**

Type: Gender.Label

Default Value: Gender.Label.NONE

Description: The gender the family name should be based on. Passing Gender.Label.NONE will generate a family name with a random gender.

- **format**

Type: Format

Default Value: Format.NONE

Description: The name format the family name should match. Passing Format.NONE will generate a family name to match a random format.

- **GetFamilyName (Origin.CategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Description: Generates a random family name based on a specific origin category.

Parameters:

- **origin**

Type: Origin.CategoryType

Description: The origin category the family name should be based on. Passing Origin.CategoryType.NONE will generate a family name with a random origin.

- **gender**

Type: Gender.Label

Default Value: Gender.Label.NONE

Description: The gender the family name should be based on. Passing Gender.Label.NONE will generate a family name with a random gender.

- **format**

Type: Format

Default Value: Format.NONE

Description: The name format the family name should match. Passing Format.NONE will generate a family name to match a random format.

- **GetFamilyName (Origin origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Description: Generates a random family name based on a specific origin.

Parameters:

- **origin**
Type: Origin
Description: The origin the family name should be based on. Passing an invalid origin will generate a family name with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the family name should be based on. Passing Gender.Label.NONE will generate a family name with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the family name should match. Passing Format.NONE will generate a family name to match a random format.

- **GetSuffix ()**

Return Type: string

Description: Generates a random name suffix.

- **GetSuffix (string origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Description: Generates a random name suffix based on a specific origin name.

Parameters:

- **origin**
Type: string
Description: The origin name the suffix should be based on. Passing "", "None", or "Any" will generate a suffix with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the suffix should be based on. Passing Gender.Label.NONE will generate a suffix with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the suffix should match. Passing Format.NONE will generate a suffix to match a random format.

- **GetSuffix (Origin.SubcategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Description: Generates a random name suffix based on a specific origin subcategory.

Parameters:

- **origin**
Type: Origin.SubcategoryType
Description: The origin subcategory the suffix should be based on. Passing Origin.SubcategoryType.NONE will generate a suffix with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the suffix should be based on. Passing Gender.Label.NONE will generate a suffix with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the suffix should match. Passing Format.NONE will generate a suffix to match a random format.

- **GetSuffix (Origin.CategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Description: Generates a random name suffix based on a specific origin category.

Parameters:

- **origin**
Type: Origin.CategoryType
Description: The origin category the suffix should be based on. Passing Origin.CategoryType.NONE will generate a suffix with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the suffix should be based on. Passing Gender.Label.NONE will generate a suffix with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the suffix should match. Passing Format.NONE will generate a suffix to match a random format.

- **GetSuffix (Origin origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**

Return Type: string

Description: Generates a random name suffix based on a specific origin.

Parameters:

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **origin**
Type: Origin
Description: The origin the suffix should be based on. Passing an invalid origin will generate a suffix with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the suffix should be based on. Passing Gender.Label.NONE will generate a suffix with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the suffix should match. Passing Format.NONE will generate a suffix to match a random format.
- **GetNickname ()**
Return Type: string
Description: Generates a random nickname.
- **GetNickname (string origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: Character
Description: Generates a random nickname based on a specific origin name.
Parameters:
 - **origin**
Type: string
Description: The origin name the nickname should be based on. Passing "", "None", or "Any" will generate a nickname with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the nickname should be based on. Passing Gender.Label.NONE will generate a nickname with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE
Description: The name format the nickname should match. Passing Format.NONE will generate a nickname to match a random format.
- **GetNickname (Origin.SubcategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random nickname based on a specific origin subcategory.
Parameters:

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **origin**
Type: Origin.SubcategoryType
Description: The origin subcategory the nickname should be based on. Passing Origin.SubcategoryType.NONE will generate a nickname with a random origin.
- **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the nickname should be based on. Passing Gender.Label.NONE will generate a nickname with a random gender.
- **format**
Type: Format
Default Value: Format.NONE
Description: The name format the nickname should match. Passing Format.NONE will generate a nickname to match a random format.
- **GetNickname (Origin.CategoryType origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random nickname based on a specific origin category.
Parameters:
 - **origin**
Type: Origin.CategoryType
Description: The origin category the nickname should be based on. Passing Origin.CategoryType.NONE will generate a nickname with a random origin.
 - **gender**
Type: Gender.Label
Default Value: Gender.Label.NONE
Description: The gender the nickname should be based on. Passing Gender.Label.NONE will generate a nickname with a random gender.
 - **format**
Type: Format
Default Value: Format.NONE
Description: The name format the nickname should match. Passing Format.NONE will generate a nickname to match a random format.
- **GetNickname (Origin origin, Gender.Label gender = Gender.Label.NONE, Format format = Format.NONE)**
Return Type: string
Description: Generates a random nickname based on a specific origin.
Parameters:
 - **origin**
Type: Origin

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Description: The origin the nickname should be based on. Passing an invalid origin will generate a nickname with a random origin.

- **gender**

Type: Gender.Label

Default Value: Gender.Label.NONE

Description: The gender the nickname should be based on. Passing Gender.Label.NONE will generate a nickname with a random gender.

- **format**

Type: Format

Default Value: Format.NONE

Description: The name format the nickname should match. Passing Format.NONE will generate a nickname to match a random format.

2.4 CNG.SETTINGS

Description: A class containing data settings for the Character Name Generator
Variables

- **GenderMaleChance**

Type: float

Default Value: GENDER_MALE_CHANCE_DEFAULT

Description: The probability of a character/name will be randomly generated as male. All gender chance settings should add up to 1.0f to function accurately.

- **GenderFemaleChance**

Type: float

Default Value: GENDER_FEMALE_CHANCE_DEFAULT

Description: The probability of a character/name will be randomly generated as female. All gender chance settings should add up to 1.0f to function accurately.

- **GenderNonBinaryChance**

Type: float

Default Value: GENDER_NON_BINARY_CHANCE_DEFAULT

Description: The probability of a character/name will be randomly generated as non-binary. All gender chance settings should add up to 1.0f to function accurately.

- **NamePrefixChance**

Type: float

Default Value: NAME_PREFIX_CHANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to have a prefix in their name. This setting should be set between 0f and 1.0f to function properly.

- **NameSuffixChance**

Type: float

Default Value: NAME_SUFFIX_CHANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to have a suffix in their name. This setting should be set between 0f and 1.0f to function properly.

- **DoubleGivenNameChance**

Type: float

Default Value: DOUBLE_GIVEN_NAME_CHANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to have two given names (e.g. Elizabeth Marie Smith). This setting should be set between 0f and 1.0f to function properly.

- **DoubleFamilyNameChance**

Type: float

Default Value: DOUBLE_FAMILY_NAME_CHANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to have two family names (e.g. Jacob Washington Smith). This setting should be set between 0f and 1.0f to function properly.

- **OriginTolerance**

Type: float

Default Value: ORIGIN_TOLERANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to match the origin of a provided or existing origin. If a matching origin is not available, a matching subcategory will be searched next. All origin tolerance settings should add to 1.0f to function accurately.

- **OriginSubcategoryTolerance**

Type: float

Default Value: ORIGIN_SUBCATEGORY_TOLERANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to match the subcategory of a provided or existing origin. If a matching subcategory is not available, a matching category will be searched next. All origin tolerance settings should add to 1.0f to function accurately.

- **OriginCategoryTolerance**

Type: float

Default Value: ORIGIN_CATEGORY_TOLERANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to match the category of a provided or existing origin. If a matching category is not available, any origin/subcategory/category will be searched next. All origin tolerance settings should add to 1.0f to function accurately.

- **OriginAnyTolerance**

Type: float

Default Value: ORIGIN_ANY_TOLERANCE_DEFAULT

Description: The probability of a character/name will be randomly generated to match any origin/subcategory/category regardless of a provided or existing origin.

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **UseUnicodeCharacters**

Type: bool

Default Value: USE_UNICODE_CHARACTERS_DEFAULT

Description: Whether or not special characters contained within names should be converted into their associated Unicode character or defaulted to their basic ASCII character.

- **PreventRecentRepeats**

Type: bool

Default Value: PREVENT_RECENT_REPEATS_DEFAULT

Description: Whether or not the generator should avoid giving a character the same given name, family name, or nickname as any other past 100 characters generated. (e.g. James “Tank” Prescott would prevent John “Tank” Smith and Jane “Inferno” Prescott but would allow Jessica “Nova” James). Disable this setting for faster performance from the generator.

Constants

- **GENDER_MALE_CHANCE_DEFAULT**

Type: float

Value: 0.45f

Description: The default value for the Gender Male Chance Setting.

- **GENDER_FEMALE_CHANCE_DEFAULT**

Type: float

Value: 0.45f

Description: The default value for the Gender Female Chance Setting.

- **GENDER_NON_BINARY_CHANCE_DEFAULT**

Type: float

Value: 0.1f

Description: The default value for the Gender Non-Binary Chance Setting.

- **NAME_PREFIX_CHANCE_DEFAULT**

Type: float

Value: 0.2f

Description: The default value for the Name Prefix Chance Setting.

- **NAME_SUFFIX_CHANCE_DEFAULT**

Type: float

Value: 0.1f

Description: The default value for the Name Suffix Chance Setting.

- **DOUBLE_GIVEN_NAME_CHANCE_DEFAULT**

Type: float

Value: 0.3f

Description: The default value for the Double Given Name Chance Setting.

- **DOUBLE_FAMILY_NAME_CHANCE_DEFAULT**

Type: float

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

Value: 0.3f

Description: The default value for the Double Family Name Chance Setting.

- **ORIGIN_TOLERANCE_DEFAULT**

Type: float

Value: 0.9f

Description: The default value for the Origin Tolerance Setting.

- **ORIGIN_SUBCATEGORY_TOLERANCE_DEFAULT**

Type: float

Value: 0.07f

Description: The default value for the Origin Subcategory Tolerance Setting.

- **ORIGIN_CATEGORY_TOLERANCE_DEFAULT**

Type: float

Value: 0.03f

Description: The default value for the Origin Category Tolerance Setting.

- **ORIGIN_ANY_TOLERANCE_DEFAULT**

Type: float

Value: 0f

Description: The default value for the Origin Any Tolerance Setting.

- **USE_UNICODE_CHARACTERS_DEFAULT**

Type: bool

Value: true

Description: The default value for the Unicode Characters Setting.

- **PREVENT_RECENT_REPEATS_DEFAULT**

Type: bool

Value: true

Description: The default value for the Recent Repeat Setting.

2.5 CNG.NAMEDATA

Description: A class for storing the name entry data loaded from JSON files.

Data Structures

- **NameEntry**

Type: struct

Description: A data structure for storing the data of each name.

Elements

- **Name**

Type: string

- **IsMale**

Type: bool

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **IsFemale**
Type: bool
- **Type**
Type: string
- **Origins**
Type: Origin []

Properties

- **IsNameDataLoaded**
Type: bool
Description: Whether or not the name data have been loaded.
- **GivenNames**
Type: NameEntry []
Description: The name entries for given names.
- **FamilyNames**
Type: NameEntry []
Description: The name entries for family names.
- **Nicknames**
Type: NameEntry []
Description: The name entries for nicknames.
- **NamePrefixes**
Type: NameEntry []
Description: The name entries for prefixes.
- **NameSuffixes**
Type: NameEntry []
Description: The name entries for suffixes.
- **Origins**
Type: Origin []
Description: The data for origins.

Functions

- **LoadNameData ()**
Return Type: void
Description: Loads the name data for the generator.

2.6 CNG.UNICODEMANAGER

Description: A class for converting stand-in special characters into either Unicode characters or ASCII characters. This class currently only supports the Latin Extended-A Unicode block.

Functions

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **SpecialCharacterCheck (string name)**
Return Type: bool
Description: Checks whether or not a name contains any special characters that need to be converted.
Parameters:
 - **name**
Type: string
Description: The name being checked for special characters.
- **ConvertSpecialCharacter (string name, bool useUnicode = true)**
Return Type: string
Description: Converts special characters into either Unicode characters or ASCII characters.
Parameters:
 - **name**
Type: string
Description: The name containing any possible special characters to convert.
 - **useUnicode**
Type: bool
Default Value: true
Description: Whether to convert special characters into Unicode characters (true) or ASCII characters (false).

2.7 CNG.JSONWRAPPER

Description: A class for reading an array of elements from a JSON file without needing to create a custom wrapper class to hold the data.

Functions

- **GetJsonFromResources (string path)**
Return Type: string
Description: Loads a JSON file from the Resources folder.
Parameters:
 - **path**
Type: string
Description: The path to the file within Resources (File extension not needed).
- **FromJson<T> (string json)**
Return Type: T []
Description: Converts a JSON string into an array of generic type parameter T.
Parameters:
 - **json**
Type: string

Description: The JSON string.

3.0 UNICODE

3.1 UNICODE, ASCII, AND SPECIAL CHARACTERS

The Unicode Manager (CNG.UnicodeManager.cs) is used for converting special characters into either Unicode characters or ASCII characters. Special characters are used in all JSON files in place of any Unicode characters in order to prevent any potential reading or display issues from Unicode characters. Currently, the Unicode Manager only supports the Latin Extended-A Unicode block.

3.2 CONVERSION TABLE

Unicode Decimal Number	Special Character	Unicode Character	ASCII Character
256	<A/>	Ā	A
257	<a/>	ā	a
258	<A^>	Ă	A
259	<a^>	ă	a
260	<A,>	Ą	A
261	<a,>	ą	a
262	<C'>	Ć	C
263	<c'>	ć	c
264	<C^>	Ĉ	C
265	<c^>	ĉ	c
266	<C.>	Č	C
267	<c.>	č	c
268	<CH>	Č	C
269	<ch>	č	c
270	<D'>	Ď	D

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

271	<d'>	đ	d
272	<D->	Ð	D
273	<d->	đ	d
274	<E/>	Ē	E
275	<e/>	ē	e
276	<E^>	Ě	E
277	<e^>	ě	e
278	<E.>	Ė	E
279	<e.>	ė	e
280	<E,>	Ǝ	E
281	<e,>	ę	e
282	<EC>	Ě	E
283	<ec>	ě	e
284	<GC>	Ĝ	G
285	<gc>	ĝ	g
286	<G^>	Ǧ	G
287	<g^>	ǧ	g
288	<G.>	Ġ	G
289	<g.>	ġ	g
290	<G,>	Ɠ	G
291	<g'>	ḡ	g
292	<H^>	Ĥ	H
293	<h^>	ĥ	h
294	<H->	Ħ	H
295	<h->	ħ	h
296	<I~>	İ	I
297	<i~>	ı	i
298	<I/>	Ī	I
299	<i/>	ī	i

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

300	<I^>	İ	I
301	<i^>	ı	i
302	<I,>	İ̇	I
303	<i,>	ı̇	i
304	<I.>	İ̈	I
305	<i.>	ı̈	i
306	<IJ>	IJ	IJ
307	<ij>	ij	ij
308	<J^>	Ĵ	J
309	<j^>	ĵ	j
310	<K,>	Ƙ	K
311	<k,>	ƙ	k
312	<kr>	ƙ	k
313	<L'>	Ł	L
314	<l'>	ł	l
315	<L,>	Ł̇	L
316	<l,>	ł̇	l
317	<LC>	Ł̈	L
318	<lc>	ł̈	l
319	<L->	Ł̣	L
320	<l->	ł̣	l
321	<L/>	Ł̤	L
322	<l/>	ł̤	l
323	<N'>	Ń	N
324	<n'>	ń	n
325	<N,>	Ñ̇	N
326	<n,>	ñ̇	n
327	<N^>	Ñ̈	N
328	<n^>	ñ̈	n

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

329	<'n>	ñ	n
330	<NG>	Ŋ	N
331	<ng>	ŋ	n
332	<O/>	Ō	O
333	<o/>	ō	o
334	<O^>	Ö	O
335	<o^>	ö	o
336	<O'>	Õ	O
337	<o'>	õ	o
338	<OE>	Œ	OE
339	<oe>	æ	oe
340	<R'>	Ŕ	R
341	<r'>	ŕ	r
342	<R,>	Ṛ	R
343	<r,>	ṛ	r
344	<R^>	Ř	R
345	<r^>	ř	r
346	<S'>	Ŝ	S
347	<s'>	ŝ	s
348	<S^>	Š	S
349	<s^>	š	s
350	<S,>	Ŝ	S
351	<s,>	ŝ	s
352	<SC>	Š	S
353	<sc>	š	s
354	<T,>	Ṭ	T
355	<t,>	ṭ	t
356	<T^>	Ť	T
357	<t^>	ť	t

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

358	<T->	Ʀ	T
359	<t->	Ƨ	t
360	<U~>	Ũ	U
361	<u~>	ũ	u
362	<U/>	Ū	U
363	<u/>	ū	u
364	<U^>	Ů	U
365	<u^>	ů	u
366	<U0>	Ů	U
367	<uo>	ũ	u
368	<U'>	Ú	U
369	<u'>	ú	u
370	<U,>	Ů	U
371	<u,>	ů	u
372	<W^>	Ẃ	W
373	<w^>	ẃ	w
374	<Y^>	Ỳ	Y
375	<y^>	ỳ	y
376	<Y'>	Ỳ	Y
377	<Z'>	Ẑ	Z
378	<z'>	ẑ	z
379	<Z.>	Ẑ	Z
380	<z.>	ẑ	z
381	<Z^>	Ž	Z
382	<z^>	ž	z
383	<ls>	ƒ	s

4.0 EXPANDING AND MODIFYING

4.1 ADDING GIVEN OR FAMILY NAMES

To expand on the lists of given names and family names for the Character Name Generator, you will need to modify the Character Name Generator - Given Names.json and Character Name Generator - Family Names.json files respectively in the Resources folder. When editing the given names JSON file, you will need to include the following elements:

- **Name**
Description: The given name.
- **Male**
Description: Whether the given name is considered a male name ("Y") or not ("N").
- **Female**
Description: Whether the given name is considered a female name ("Y") or not ("N").
- **Origins**
Description: A comma separated list of origin names associated with the given name. The value "Any" can be used to allow the given name to be used with any origin.

When editing the family names JSON file, you will need to include the following elements:

- **Name**
Description: The family name.
- **Origins**
Description: A comma separated list of origin names associated with the family name. The value "Any" can be used to allow the family name to be used with any origin.

If any given names or family names that are added contain Unicode characters, be sure replace the Unicode characters with the associated special characters (see 3.2 Conversion Table).

4.2 ADDING NAME PREFIXES OR SUFFIXES

To expand on the lists of name prefixes and suffixes for the Character Name Generator, you will need to modify the Character Name Generator - Prefixes.json and Character Name Generator - Suffixes.json files respectively in the Resources folder. When editing the name prefixes JSON file, you will need to include the following elements:

- **Name**
Description: The name prefix.

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

- **Male**
Description: Whether the name prefix is considered a male prefix (“Y”) or not (“N”).
- **Female**
Description: Whether the name prefix is considered a female prefix (“Y”) or not (“N”).
- **Origins**
Description: A comma separated list of origin names associated with the name prefix. The value “Any” can be used to allow the name prefix to be used with any origin.

When editing the name suffixes JSON file, you will need to include the following elements:

- **Name**
Description: The name suffix. Starting the suffix with a comma will prevent the full name from placing a space before the suffix.
- **Origins**
Description: A comma separated list of origin names associated with the name suffix. The value “Any” can be used to allow the name suffix to be used with any origin.

If any name prefix or suffix that are added contain Unicode characters, be sure replace the Unicode characters with the associated special characters (see 3.2 Conversion Table).

4.3 ADDING NICKNAMES

To expand on the lists of nicknames for the Character Name Generator, you will need to modify the Character Name Generator - Nicknames.json file in the Resources folder. When editing the nicknames JSON file, you will need to include the following elements:

- **Name**
Description: The nickname.
- **Male**
Description: Whether the nickname is considered a male name (“Y”) or not (“N”).
- **Female**
Description: Whether the nickname is considered a female name (“Y”) or not (“N”).
- **Type**
Description: The broad category of the nickname (e.g. History, Mythology, etc.). The value “General” can be used to allow the nickname to be used with any type, however the Character Name Generator does not filter by type in the current version.
- **Origins**
Description: A comma separated list of origin names associated with the nickname. The value “Any” can be used to allow the nickname to be used with any origin.

Character Name Generator

Version 1.0.0

Copyright © 2019

Flight Paper Studio LLC

If any nicknames that are added contain Unicode characters, be sure replace the Unicode characters with the associated special characters (see 3.2 Conversion Table).

4.4 ADDING GENDERS AND PRONOUNS

To expand on the supported genders for the Character Name Generator, you will need to modify the following:

- Gender.Label
- Gender.ToString (Label gender, bool isAbbreviated = false)
- NameGenerator.GetGender ()
- NameGenerator.GetPronouns (Gender.Label gender)
- NameGenerator.FilterByGender (NameData.NameEntry data, Gender.Label gender)

To expand on the supported pronouns for the Character Name Generator, you will need to modify the following:

- Gender.Pronouns
- Gender.GetSubjectivePronoun (Pronouns pronoun, bool isCapitalized = true)
- Gender.GetObjectivePronoun (Pronouns pronoun, bool isCapitalized = true)
- Gender.GetPossessivePronoun (Pronouns pronoun, bool isCapitalized = true)
- NameGenerator.GetPronouns (Gender.Label gender)

4.5 ADDING NAME FORMATS

To expand on the supported name formats for the Character Name Generator, you will need to modify the following:

- NameGenerator.Format
- NameGenerator.GetFormat ()