

# family-tree

family tree utility CLI app

## Synopsis

An utility tool to create and view family tree records.

# family-tree add

adds persons and roles

## Synopsis

Subcommand to add names of family members and roles of them. For example: // adds a person in family

```
family-tree add person john
```

// adds an role entry e.g father, mother, son, daughter

```
family-tree add role father
```

```
family-tree add [flags]
```

## Options

```
-h, --help  help for add
```

# family-tree add person

adds person

## Synopsis

Subcommand to add persons (duplicate not allowed). For example: // adds a person in family

```
family-tree add person william
```

## Options

```
-h, --help    help for person
```

# family-tree add role

adds role

## Synopsis

Subcommand to add roles (duplicate not allowed). For example: // adds role of a person // allowed roles [father, mother, son, daughter, brother, sister, wife, husband]

```
family-tree add role father
```

```
family-tree add role son
```

```
family-tree add role [flags]
```

## Options

```
-h, --help    help for role
```

# family-tree connect

connects two person

## Synopsis

Subcommand to connect two persons with a relationship. For example: // connects two persons of a family

```
family-tree connect william --as brother --of john
```

// default gender of person2 is male if it isn't specify it with --person2-gender or -g

```
family-tree connect adam --as brother --of mary --person2-gender female
```

```
family-tree connect [flags]
```

## Options

<code>-a, --as string</code>	specifies the role of the person
<code>-h, --help</code>	help for connect
<code>-o, --of string</code>	specifies the relation with person2
<code>-g, --person2-gender string</code>	specifies gender of person2 (default "male")

# family-tree count

counts relatives

## Synopsis

Subcommand to count relatives by their roles. For example: `// count sons`

```
family-tree count sons --of william
```

`// count daughters`

```
family-tree count daughters --of john
```

```
family-tree count [flags]
```

## Options

<code>-h, --help</code>	help for count
<code>-o, --of string</code>	specifies the person
<code>-g, --person-gender string</code>	specifies person gender (default "male")

# family-tree father

shows father name

## Synopsis

Subcommand to find father name. For example:

```
family-tree father --of william
```

```
family-tree father [flags]
```

## Options

<code>-h, --help</code>	help for father
<code>-o, --of string</code>	specifies person name