

Zadatak 1.

U droolsu implementirati prologov geneaology primer:
<http://kti.mff.cuni.cz/~bartak/prolog/genealogy.html>

Za ulazne podatke iskoristiti:

```
// grand parents
```

```
ksession.insert(new Man("john"));
```

```
ksession.insert(new Woman("janet"));
```

```
// parent
```

```
ksession.insert(new Man("adam"));
```

```
ksession.insert(new Parent("john", "adam"));
```

```
ksession.insert(new Parent("janet", "adam"));
```

```
ksession.insert(new Man("stan"));
```

```
ksession.insert(new Parent("john", "stan"));
```

```
ksession.insert(new Parent("janet", "stan"));
```

```
// grand parents
```

```
ksession.insert(new Man("carl"));
```

```
ksession.insert(new Woman("tina"));
```

```
//
```

```
// parent
```

```
ksession.insert(new Woman("eve"));
```

```
ksession.insert(new Parent("carl", "eve"));
```

```
ksession.insert(new Parent("tina", "eve"));
```

```
//
```

```
// parent
```

```
ksession.insert(new Woman("mary"));
```

```
ksession.insert(new Parent("carl", "mary"));
```

```
ksession.insert(new Parent("tina", "mary"));
```

```
ksession.insert(new Man("peter"));
ksession.insert(new Parent("adam", "peter"));
ksession.insert(new Parent("eve", "peter"));
```

```
ksession.insert(new Man("paul"));
ksession.insert(new Parent("adam", "paul"));
ksession.insert(new Parent("mary", "paul"));
```

```
ksession.insert(new Woman("jill"));
ksession.insert(new Parent("adam", "jill"));
ksession.insert(new Parent("eve", "jill"));
```

Rezultati treba da budu sledeci:

-----Woman-----

janet

tina

jill

mary

eve

-----Man-----

peter

stan

carl

john

adam

paul

-----Father-----

carl, mary

carl, eve

john, stan

john, adam

adam, jill

adam, paul

adam, peter

-----Mother-----

janet, stan

janet, adam

tina, mary

tina, eve

mary, paul

eve, jill

eve, peter

-----Son-----

peter, eve

peter, adam

stan, john

stan, janet

adam, john

adam, janet

paul, mary

paul, adam

-----Daughter-----

jill, eve

jill, adam

mary, tina

mary, carl

eve, tina

eve, carl

-----Siblings-----

stan, adam

adam, stan

jill, peter

peter, jill

mary, eve

eve, mary

jill, peter

jill, paul

paul, peter

paul, jill

peter, paul

peter, jill

stan, adam

adam, stan

mary, eve

eve, mary

-----FullSiblings-----

stan, adam

adam, stan

jill, peter

peter, jill

mary, eve

eve, mary

jill, peter

peter, jill

stan, adam

adam, stan

mary, eve

eve, mary

-----Uncle-----

stan, jill

stan, paul

stan, peter

stan, jill

stan, paul

stan, peter

-----Aunt-----

eve, paul

eve, paul

mary, jill

mary, peter

mary, jill

mary, peter

-----GrantParents-----

carl, jill

carl, peter

carl, paul

janet, jill

janet, paul

janet, peter

tina, jill

tina, peter

tina, paul

john, jill

john, paul

john, peter

Zadatak 2.

Prosiriti model primera sa proslih casova sa generickom classom `IsPartOf<T>` koja se sastoji od dva poziciona argumenta `whole` i `part`. Napraviti pravilo koje daje 5 posto popusta na porudzbine koje sadrze povezane Item-e. Npr. ako znamo da Item `"HomeCinema"` sadrzi `"RemoteControl"` (`IsPartOf("HomeCinema", "RemoteControl")`) i ako znamo da Item `"RemoteControl"` sadrzi `"Battery"` (`IsPartOf("RemoteControl", "Battery")`), u slucaju ako korisnik u svojoj porudzbini ima `HomeCinema` i `Battery`, toj porudzbini dodati popust od 5%.