

Public OWLOntology addIndividual(String ID, String typeName, String propName, OWLIndividual indv)

ID koduyla typeName isimli sınıfın bir individual'ını yaratır. Bu individual'ın propName isimli objectProperty'sinin değerini parametre olarak aldığı indv ile doldurur. Elde ettiği yeni ontoloji yapısını sabit diske kaydeder ve dönüş değeri olarak döndürür.

public OWLOntology removeIndividuals(String TypeName)

Tipi parametre olarak elde edilen typeName isminde OWLClass olan tüm individual'ların silinmesi işleminde sorumludur. Elde ettiği yeni ontoloji yapısını sabit diske kaydeder ve dönüş değeri olarak döndürür.

Public OWLOntology removeAxiom(OWLIndividual ind, OWLObjectProperty prop)

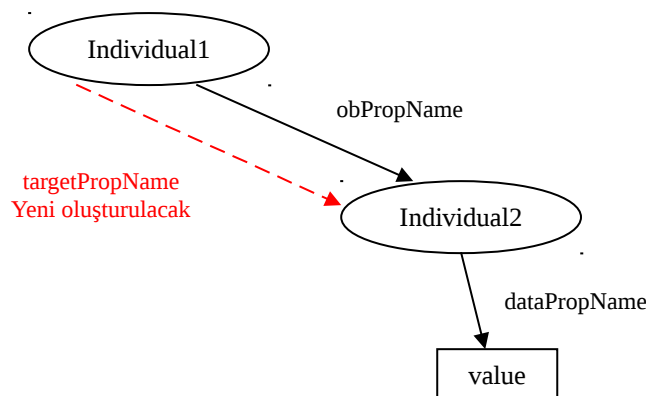
Parametre olarak aldığı OWLIndividual'ın yine parametre olarak aldığı OWLObjectProperty ile bağlı olduğu tüm individual'ların silinmesinden sorumludur. Örneğin *kutuphaneX* kodlu individual'ın *sahiptir* özelliğinin değeri olan tüm individual'ların silinmesi gibi. Elde ettiği yeni ontoloji yapısını sabit diske kaydeder ve dönüş değeri olarak döndürür.

Public void markProperIndividuals(String typeName, String propertyName, String dataPropName)

Parametre olarak aldığı typeName ismindeki sınıf tipinde olan ve propertyName ismindeki ObjectProperty'e sahip olan tüm individual'ların dataPropName isimli dataProperty'sinin değerini true yapan SWRL kuralını Pellet çıkarsama motoru aracılığıyla işletir. Elde ettiği genişletilmiş yeni ontolojiyi başka bir isimle sabit diske kaydeder.

Public void extendOntology(String typeName, String obPropName, String dataPropName, int value, String targetPropName)

typeName tipinde ve obPropName ismindeki objectProperty'sinin değeri olan individual'ların dataPropName ismindeki dataTypeProperty'lerinin değeri parametre olarak alınan value değerine eşit olan idividual'lar ile bu objectProperty aracılığıyla ilişkili olduğu individualları arasında targetPropName isimli objectProperty axiomu ekleyen SWRL kuralını pellet çıkarsama motoru aracılığıyla işletir. Elde ettiği genişletilmiş yeni ontolojiyi başka bir isimle sabit diske kaydeder. (Aşağıdaki şekildeki örneğe bakınız...)



public static void main()

Gerçekleştirdiğiniz diğer metotları uygun giriş değerlerini parametre olarak göndererek test eden kodu içerir.

Not: Tüm metotları üzerinde çalışacağı ontolojiyi bilen ve fiziksel adresini sabit olarak tutan bir sınıf içerisinde gerçekleştiriniz. Manager oluşturma ve ontoloji yükleme gibi ortak işleri yapıcı metot içerisinde gerçekleştirebilirsiniz.