ObjectProperty

* Functional: Khẳng định rằng thuộc tính được chọn là Functional. Về mặt trực quan, điều này có nghĩa là đối với bất kỳ một individual nào, thuộc tính có thể có nhiều nhất một giá trị. Nếu nhiều individual được chỉ định làm giá trị cho thuộc tính, thì các giá trị này sẽ được suy luận để biểu thị cùng một đối tượng.
* Inverse Functional: Khẳng định rằng thuộc tính đã chọn mà Inverse Functional. Trực quan, điều này có nghĩa là thuộc tính nghịch đảo của thuộc tính đã chọn là Functional.
* Transitive: Khẳng định một thuộc tính đã chọn là Transitive (Bắc cầu). Nghĩ là nếu một cá thể x được liên kết với cá thể y, và cá thể y được liên kết với cá thể z, x cũng sẽ được liên kết với cá thể z.
* Symmetric: Khẳng định thuộc tính được chọn là Symmetric (Đối xứng). Có nghĩa là thuộc tính đó chính là nghịch đảo của chính nó, vậy nếu một cá thể x được liên kết với cá thể y thì cá thể y cũng được liên kết với cá thể x bằng chính thuộc tính đó.
* Asymmetric: Khẳng định thuộc tính được chọn là Asymmetric (Bất đối xứng). Nghĩa là nếu cá thể x được liên kết với cá thể y thì cá thể y không được liên kết với x bằng chính thuộc tính đó.
* Reflexive: Khẳng định thuộc tính được chọn là Reflexive (Phản chiếu). Khẳng định một thuộc tính là Reflexive khiến mọt cá thể đều được liên kết đến chính nó thông qua thuộc tính đó.
* Irreflexive: Khẳng định thuộc tính được chọn là Irreflexive (Bất phản chiếu). Việc khẳng định một thuộc tính là Irreflexive nghĩa là một cá thể không thể kết nối với chính nó thông qua thuộc tính đó.

Class

* Equivalent To: Các lớp tương đương với lớp được chọn.
* SubClass Of: Các lớp cha của lớp hiện tại.
* General Class Axioms: Hiển thị General Class Axiom (Tiên đề lớp chung) đề cập đến lớp được chọn.
* Instances: Hiển thị các cá thể mà có lớp được chọn làm kiểu của nó trong tiên đề khẳng định lớp.
* Disjoint With: Hiển thị danh sách các biểu thức lớp không có phần giao với lớp được chọn.
* Target for Key: Chỉ định một danh sách hỗn hợp các thuộc tính đối tượng và dữ liệu hoạt động như một khóa cho các phiên bản của lớp được chọn. Các khóa là một tính năng mới trong OWL2 và bao gồm một tập hợp các thuộc tính. Đối với một cá thể nhất định, các giá trị cụ thể của các thuộc tính này được kết hợp với nhau chỉ ra tính khác biệt. Ví dụ, một khóa bao gồm hasSurname và hasDateOfBirth có thể được sử dụng để chỉ ra tính khác biệt của các cá thể trong lớp Person.
* Disjoint Union Of: Chỉ định rằng lớp được chọn là lớp chính trong teien đề lớp DisjointUnion.
* SubClassOf (Anonymous Ancestor): Protégé kiểm tra tổ tiên của lớp đã chọn và tích lũy tất cả các lớp cha của chúng sau đó hiển thị trong phần này.