

ValueConcept
name
__and__(value: typing.Self): typing.Self __hash__(): int __init__(c_type: ConceptType, role: str, value: typing.Any): None __neg__(): Concept __or__(value: typing.Self): typing.Self at_least_value(role: str, o: typing.Any): typing.Self at_most_value(role: str, o: typing.Any): typing.Self clone(): typing.Self compute_atomic_concepts(): set[Concept] compute_name(): typing.Optional[str] exact_value(role: str, o: typing.Any): typing.Self get_roles(): set[str] replace(a: Concept, c: Concept): Concept

Concept
DEFAULT_NAME : str SPECIAL_STRING : str _name : str _type : ConceptType name num_new_concepts : int type
__and__(value: typing.Self): typing.Self __eq__(value: typing.Self): bool __iand__(value: typing.Self): typing.Self __init__(c_type: ConceptType, name: str): None __ior__(value: typing.Self): typing.Self __irshift__(value: typing.Self): typing.Self __ne__(value: typing.Self): bool __or__(value: typing.Self): typing.Self __rshift__(value: typing.Self): typing.Self __str__(): str is_atomic(): bool is_complemented_atomic(): bool

HasValueInterface
_value : Any value
__init__(role: str, value: typing.Any): None