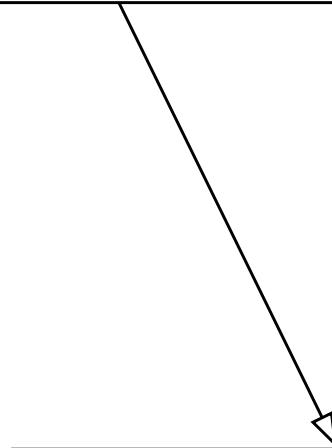


AllSomeConcept
_name : str
__hash__(): int __init__(role: str, c: Concept, c_type: ConceptType): None __neg__(): Concept all(role: str, concept: Concept): typing.Self clone(): typing.Self compute_atomic_concepts(): set[Concept] compute_name(): str get_atoms(): list[typing.Self] get_roles(): set[str] is_complemented_atomic(): bool new(c_type: ConceptType, role: str, concept: Concept): typing.Self replace(a: Concept, c: Concept): Concept some(role: str, concept: Concept): typing.Self



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



HasRoleConceptInterface
__init__(role: str, concept: Concept): None