EERD (Enhanced Entity Relationship Diagram):

EERDs are specialized ERDs that can be extremely useful for modeling a database. EERDs use concepts that are closely related to OOP and object-oriented design.

The concepts are:

1. Subtypes and Supertypes.
2. Specialization and Generalization.
3. Categories and Union Types.
4. Attribute Inheritance.
5. Overlapping and Disjoint Constraints.
6. Multivalued Attributes.
7. Derived Attributes.
8. Relationship Attributes.
9. Constraints and Special Relationships.
10. Aggregation.

	ERD	EERD
Basic Structure	Represents entities, attributes, and relationships.	Extends ERD concepts with subtypes, supertypes, categories, union types, attribute inheritance.
Hierarchical Relationships	No direct support for subtypes and specialization/generalization.	Supports subtypes, supertypes, specialization, generalization, overlapping/disjoint constraints.
Inheritance of Attributes	Attributes are specific to individual entities.	Subtypes can inherit attributes from supertypes.
Special Relationships	No support for identifying relationships or recursive relationships.	Supports identifying relationships, recursive relationships.
Multivalued Attributes	Limited support for multivalued attributes.	Supports multivalued attributes.
Derived Attributes	Limited support for derived attributes.	Supports derived attributes.
Relationship Attributes	Relationships have no associated attributes.	Supports attributes associated with relationships.
Whole-Part Relationships	Limited support for aggregation.	Supports aggregation to represent whole-part relationships.
Constraint Flexibility	Limited constraint support	Provides more constraint options
Use Cases	Well-suited for simpler database designs.	Better suited for complex data models with diverse relationships and attributes.

EERD is better than ERD because EERD supports more types of attributes, supports more types of relationships, more flexible, and can be used for complex data models.