

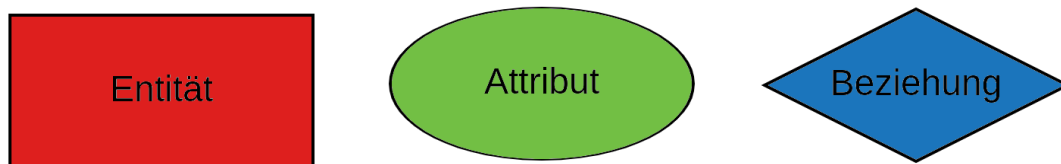
# Das Entity-Relationship-Modell

Das Entity-Relationship-Modell – abgekürzt mit ER-Modell oder ERM – dient als Grundlage für einen Datenbankentwurf. Bevor mittels SQL angefangen wird, Tabellen und Beziehungen anzulegen oder gar Daten einzufügen, zu bearbeiten und abzufragen, wird zuerst mit einem ER-Modell geplant, wie die Datenbank aufgebaut sein soll.

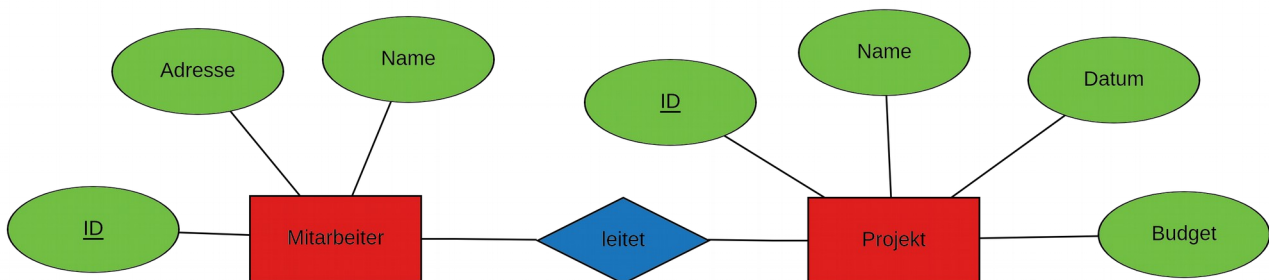
Der Einsatz von ER-Modellen ist in der Praxis ein gängiger Standard für die Datenmodellierung. Mithilfe des ER-Modells soll eine Typisierung von Objekten, ihrer Beziehungen untereinander und deren Attribute, stattfinden. Gute Datenbankentwickler können in kurzer Zeit sehr komplexe ER-Modelle verstehen und umsetzen.

Um was genau es sich bei diesen Elementen handelt, klären die folgenden Punkte:

- Eine **Entität** ist ein Objekt der Wirklichkeit, wie z.B. eine Person oder ein Auto.
- Eine **Beziehung** ist ein Zusammenhang zwischen zwei oder mehr Entitäten, wie z.B. „Person wohnt in Ort“ oder „Civic wurde gebaut von Honda“.
- Ein **Attribut** ist eine Eigenschaft von einer Entität, wie z.B. der Name einer Person oder die Anzahl Räder eines Autos.



## Beispiel für ein (einfaches) ER-Modell



Es gibt die Entitäten Mitarbeiter und Projekt. Ein Mitarbeiter hat als Attribut nur seinen Namen. Ein Projekt hat als Attribute einen Namen, ein Datum und ein Budget. Die einzige Beziehung in diesem Modell gibt an, dass ein Mitarbeiter Projekte leitet.

# Arbeitsaufträge

## Aufgabe 1:

Erstellt ein ER-Modell von der Shop-Datenbank mit den unten angegebenen Anforderungen.

### Anforderungskatalog für den Online-Shop (Grundversion):

- Jeder Kunde hat eine ID, einen Namen und eine Adresse.
- Ein Kunde kann Bestellungen aufgeben.
- Jede Bestellung hat eine ID und ein Datum.
- Jede Bestellung besteht aus einzelnen Posten.
- Jeder Posten hat eine ID und eine Menge.
- Zu jedem Posten gehört ein Produkt.
- Jedes Produkt hat eine ID, einen Namen, eine Einheit und einen Preis.

### Anforderungskatalog für den Online-Shop (Erweiterte Version, für Schnelle):

- Jedes Produkt kann zu mehreren Kategorien gehören.
- Jede Kategorie hat eine ID, einen Namen und eine Beschreibung.
- Jedes Produkt wird von einem Lieferanten geliefert.
- Jeder Lieferant hat eine ID, einen Namen, eine Adresse und eine Telefonnummer.
- Jede Bestellung wird von einem Mitarbeiter verwaltet.
- Jeder Mitarbeiter hat eine ID, einen Namen, ein Geburtsdatum, ein Foto und Notizen.
- Jede Bestellung wird von einem Spediteur versandt.
- Jeder Spediteur hat eine ID, einen Namen und eine Telefonnummer.

## Aufgabe 2:

Vergleicht euer ER-Modell mit der tatsächlichen Datenbank. Diese findet ihr unter [https://www.w3schools.com/SQL/trysql.asp?filename=trysql\\_select\\_all](https://www.w3schools.com/SQL/trysql.asp?filename=trysql_select_all).

Beantwortet die folgenden Fragen in einer Datei im Gruppenordner:

- a) Welche Attribute wurden in der Datenbank anders als im Modell umgesetzt? Stelle eine Vermutung auf, warum das so sein könnte!
- b) Wie wurden die Beziehungen in der Datenbank umgesetzt? Beschreibe das Vorgehen!

## Aufgabe 3:

Überlegt euch, wie die Datenbank für eure eigene Website aussehen könnte.

- a) Erstellt einen Anforderungskatalog wie in Aufgabe 1.
- b) Identifiziert die Entitäten, die Attribute und die Beziehungen.
- c) Erstellt ein ER-Modell.