Dietrich Mosel

Professor Labouseur

Database Management

November 20, 2017

Lab 9: Normalization 3

1. Identify and document all functional dependencies

Functional Dependencies for the People Table…

{PID} → {FirstName, LastName, Age}

Functional Dependencies for the Astronauts Table…

{PID} → {YearsFlying, GolfHandicap, SpouseName}

Functional Dependencies for the Engineers Table…

{PID} → {HighestAcademicDegreeEarned, FavoriteVideoGame}

Functional Dependencies for the FlightControlOperators Table…

{PID} → {ChairPreference, PreferredDrink, RecommendedHangoverCure}

Functional Dependencies for the Crew Table…

{PID} → {SCID}

Functional Dependencies for the Spacecraft Table…

{SCID} → {Name, TailNumber, WeightInTons, FuelType, CrewCapacity}

Functional Dependencies for the SpacecraftSystems Table…

{SCID} → {SYSID}

Functional Dependencies for the Systems Table…

{SYSID} → {Name, Description}

Functional Dependencies for the SystemParts Table…

{SYSID} → {PartID, Name, Description, CostUSD}

Functional Dependencies for the Parts Table…

{PartID} → {Name, Description}

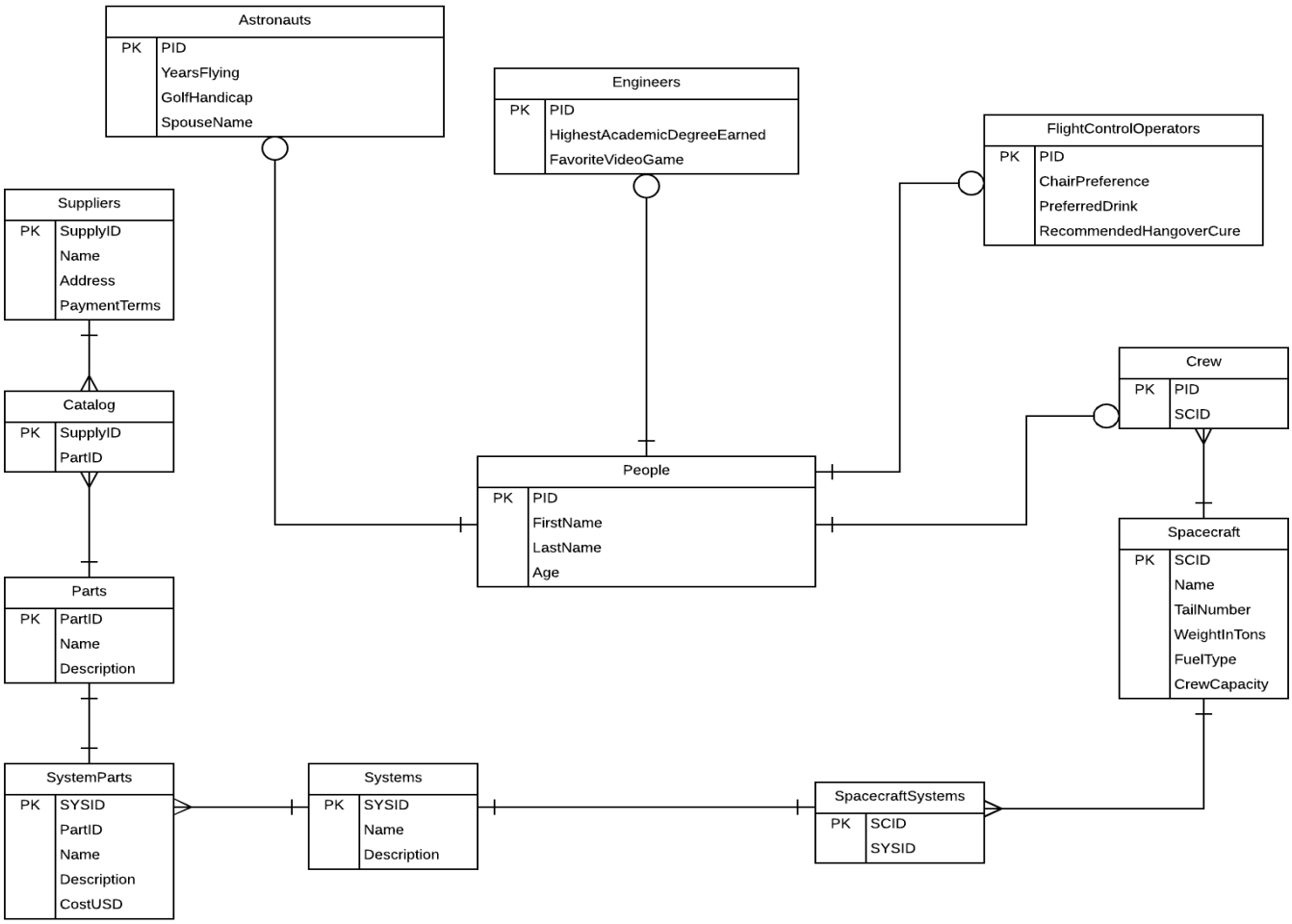
Functional Dependencies for the Catalog Table…

{SupplyID} → {PartID}

Functional Dependencies for the Suppliers Table…

{SupplyID} → {Name, Address, PaymentTerms}

2. Draw a fully annotated E/R diagram



3. Convince me that your database is in 3NF (or even better, in Boyce-Codd Normal Form)

My database is in at least 3NF because there are no multi-key dependencies for one. BCNF is stricter than 3NF in that each element must distinguish something about the key in some way. This database is also in BCNF because there are no repeating values or redundancy in any table. The “people” table separated a lot of excess data while the “SystemParts” table serves as an intermediary between “Parts” and “Systems.”