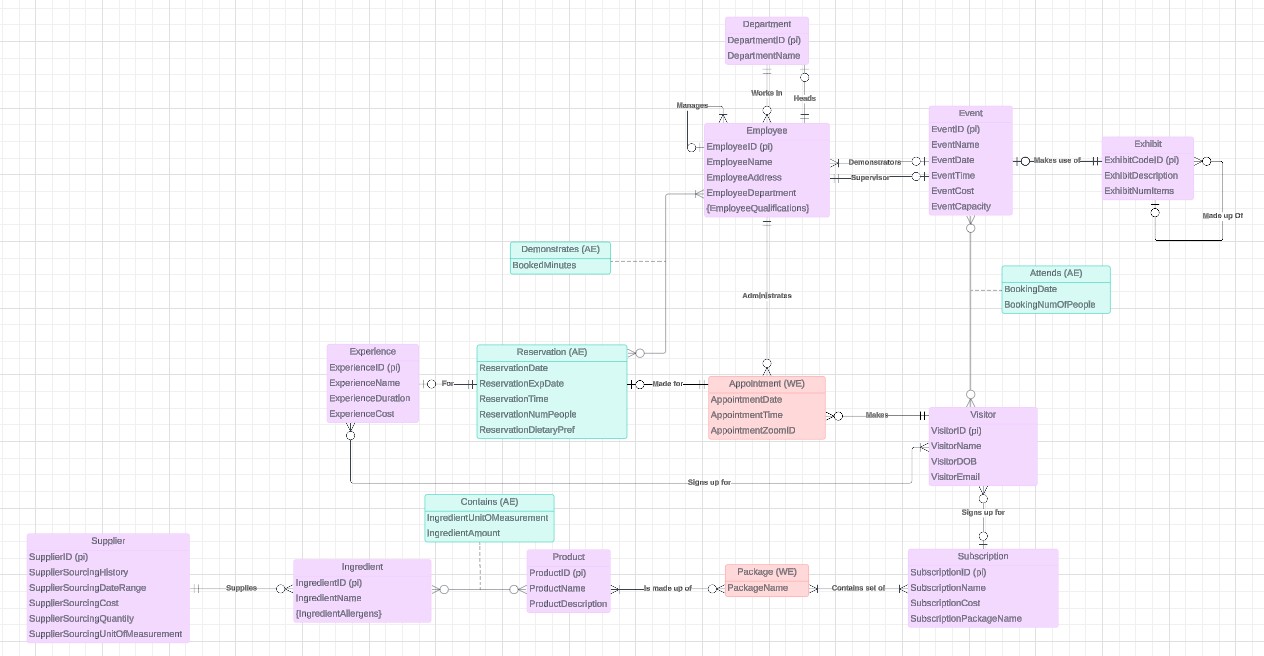
**Case Background**

Case Background Chocolate Nation is a Belgian chocolate museum that offers opportunities for hands-on events and tasting experiences to an exclusive audience. They are expanding to Australia, and the project manager has enlisted you as a database designer to create a database design (including designing the ERD, transforming, and creating the tables) and run some analysis for the Australian branch. Please note this is a hypothetical scenario, so please do not contact the museum. Chocolate Nation provides different opportunities for visitors to explore the history of chocolates. There is a wide range of options for visitors to book during their visit to the museum. Every visitor must be registered and identified with a visitor identifier number. Other details of a visitor include the name, date of birth, and email. Some events happen throughout the year. Every event is uniquely identified. Other details of the event include name, date, time, cost, and capacity. Events can be one-off or recurring. An event will make use of a maximum of one exhibit that the museum owns. A code identifies every exhibit. The details of the exhibit, such as the description and number of items, are stored. An exhibit may be made up of multiple exhibits. When making a booking for an event, the booking date and the number of people the booking is for are recorded as well. Events are run by multiple employees (one supervisor and multiple demonstrators, but at least one). A visitor can also sign up for a tasting experience. Every experience offered is uniquely identified. Other details of the experience include name, duration, and cost. Experiences are offered every day. To schedule an experience, the visitor must make an appointment with an employee (who is an administrator) for a particular date and time. The appointment by itself doesn’t cost anything, and a Zoom meeting ID is sent to the visitor. After the appointment, if the visitors want to proceed with the booking, a reservation for the experience is created. A reservation is made for an experience with three employees (who are all demonstrators) and for a visitor. Other details of the reservation, such as the employee making the reservation, the date of reservation, the experience date, time, the number of people, and the group’s different dietary preferences (dairy-free, nut-allergy, etc), are stored. The demonstrators are booked for a certain number of minutes, which is also recorded. Chocolate Nation records details of their employees, each identified by a unique identifier. Other details captured include the name, address, the department they work for and qualifications. Each employee can have only one direct manager. The department is identified with a unique identifier, and the department name is recorded. An employee (the head of the department) can head only one department, and every department has one employee who is the head of the department. The products are identified by a unique identifier, and their name and description are recorded. Products can contain multiple ingredients; an ingredient can be used in many products. Each ingredient is identified by a unique identifier alongside the name and allergens. The amount and the unit of measurement of ingredients in every product are recorded as well. Every ingredient is sourced from a single supplier. Suppliers may change, and hence, the history of sourcing must be stored. The date range of the sourcing must be recorded for documentation and food safety reasons. The cost, quantity, and unit of measurement (for example, $5 for one kilo (or) $50 per 250 grams) is also recorded in the sourcing. Visitors can sign up for one subscription from the range of subscriptions that Chocolate Nation offers. The name and cost of every subscription will be recorded. Every subscription has a set of packages that are ready for visitors to pick up. The identification number for the package is repeated for every subscription it is under. The name of the package is stored. A package is made up of multiple products. A product may belong to multiple packages.

**Task 1: ER Diagram**

**Task 2: Logical Transformation**

**Step 1: Strong Entities** - For each entity in the ER model, create a relation (i.e. a table that includes all the simple attributes. Make sure to identify the primary key for the relation (i.e. the PK of the entity). If there is a composite attribute, you can expand them. Leave multi-valued attributes out

Employee(EmployeeID(pk), EmployeeName, EmployeeAddress, EmployeeDepartment)

Department(DepartmentID(pk), DepartmentName)

Event(EventID(pk), EventName, EventDate, EventTime, EventCost, EventCapacity)

Exhibit(ExhibitCodeID(pk), ExhibitDescription, ExhibitNumItems)

Experience(ExperienceID(pk), ExperienceName, ExperienceDuration, ExperienceCost)

Visitor(VisitorID(pk), VisitorName, VisitorDOB, VisitorEmail)

Subscription(SubscriptionID(pk), SubscriptionName, SubscriptionCost,

SubscriptionPackageName)

Product(ProductID(pk), ProductName, ProductDescription)

Ingredient(IngredientID(pk), IngredientName)

Supplier(SupplierID(pk), SupplierSourcingHistory, SupplierSourcingDateRange,

SupplierSourcingCost, SupplierSourcingQuantity,

SupplierSourcingUnitOfMeasurement)

**Step 2: Weak Entities** - For each weak entity in the ER model, create a relation that includes all the simple attributes. The primary key of the relation is the combination of the primary key/s of the 'owner' and the main attribute of the weak entity itself

Appointment(AppointmentZoomID(pk), VisitorID(pk,fk), EmployeeID(pk,fk),

AppointmentDate, AppointmentTime)

Package(PackageName(pk), SubscriptionID(pk,fk), ProductID(pk,fk))

**Step 3: 1:1 Relationship** - For each 1:1 relationship identify the two relations corresponding to the entities participating in the relationship. Choose the PK of the Relation (usually the one with the mandatory constraint) and make it the foreign key of the other relation.

Department(DepartmentID(pk), DepartmentName, EmployeeID(fk))

Employee(EmployeeID(pk), EmployeeName, EmployeeAddress,

EmployeeDepartment, DepartmentID(fk), EventID(fk))

Event(EventID(pk), EventName, EventDate, EventTime, EventCost, EventCapacity, EmployeeID(fk), ExhibitID(fk))

Exhibit(ExhibitCodeID(pk), ExhibitDescription, ExhibitNumItems, EventID(fk))

**Step 4: 1:N relationship** - For each binary 1 to N relationship identify the relations that represent the participating entity at the N (i.e. many) side of the relationship. Include as the foreign key in the relation that holds the N side, the primary key of the other entity (that holds the 1 side)

Ingredient(IngredientID(pk), IngredientName, SupplierID(fk))

Employee(EmployeeID(pk), EmployeeName, EmployeeAddress,

EmployeeDepartment, DepartmentID(fk), EventID(fk), ManagerID(fk, references EmployeeID))

Exhibit(ExhibitCodeID(pk), ExhibitDescription, ExhibitNumItems, EventID(fk),

ParentExhibitID(fk, references exhibitID))

Appointment(AppointmentZoomID(pk), VisitorID(pk,fk), EmployeeID(pk,fk),

AppointmentDate, AppointmentTime, VisitorID(fk))

Visitor(VisitorID(pk), VisitorName, VisitorDOB, VisitorEmail, SubscriptionID(fk))

**Step 5: M:N relationship** - For each binary M:N Relationship create a new relation to represent the relationship. The primary key of the new relation is the combination of the primary keys of the two connected entities. This is an associative entity. If there are any attributes on the relationship, then include them.

Demonstrates(EmployeeID(pk,fk), BookedMinutes)

Attends(EventID(pk,fk), VisitorID(pk,fk), BookingDate, BookingNumOfPeople)

Contains(IngredientID(pk,fk), ProductID(pk,fk), IngredientUnitOfMeasurement, IngredientAmount)

PackagedProduct(PackageName(pk,fk), SubscriptionID(pk,fk), ProductID(pk,fk))

SubscriptionPackage(PackageName(pk,fk), SubscriptionID(pk,fk), ProductID(pk,fk))

SignUpFor(VisitorID(pk,fk), ExperienceID(pk,fk))

**Step 6: Multi-valued attributes** - For each multivalued attribute, create a new relation that includes the multivalued attribute and the primary key of the entity where the multivalued attribute is attached.

IngredientAllergens(IngredientID(pk,fk), Allergen(pk))

EmployeeQualifications(EmployeeID(pk,fk), Qualification(pk))

**Step 7: Associative entities -** For each n-ary relationship create a new relation to represent the relationship. The primary key of the new relation is the combination of the primary keys of the participating entities that hold the N (many) side. In most cases of an n-ary relationship, all the participating entities hold a many side.

Reservation(ReservationID(pk), EmployeeID(fk), ExperienceID(fk),

AppointmentZoomID(fk), ReservationDate, ReservationExpDate, ReservationTime,

ReservationNumPeople, ReservationDietaryPref)

**End result:**

Experience(ExperienceID(pk), ExperienceName, ExperienceDuration, ExperienceCost)

Subscription(SubscriptionID(pk), SubscriptionName, SubscriptionCost,

SubscriptionPackageName)

Product(ProductID(pk), ProductName, ProductDescription)

Supplier(SupplierID(pk), SupplierSourcingHistory, SupplierSourcingDateRange,

SupplierSourcingCost, SupplierSourcingQuantity, SupplierSourcingUnitOfMeasurement)

Department(DepartmentID(pk), DepartmentName, EmployeeID(fk))

Event(EventID(pk), EventName, EventDate, EventTime, EventCost, EventCapacity,

EmployeeID(fk), ExhibitID(fk))

Ingredient(IngredientID(pk), IngredientName, SupplierID(fk))

Employee(EmployeeID(pk), EmployeeName, EmployeeAddress,

EmployeeDepartment, DepartmentID(fk), EventID(fk), ManagerID(fk, references EmployeeID))

Exhibit(ExhibitCodeID(pk), ExhibitDescription, ExhibitNumItems, EventID(fk),

ParentExhibitID(fk, references exhibitID))

Appointment(AppointmentZoomID(pk), VisitorID(pk,fk), EmployeeID(pk,fk),

AppointmentDate, AppointmentTime, VisitorID(fk))

Visitor(VisitorID(pk), VisitorName, VisitorDOB, VisitorEmail, SubscriptionID(fk))

Demonstrates(EmployeeID(pk,fk), BookedMinutes)

Attends(EventID(pk,fk), VisitorID(pk,fk), BookingDate, BookingNumOfPeople)

Contains(IngredientID(pk,fk), ProductID(pk,fk), IngredientUnitOfMeasurement, IngredientAmount)

PackagedProduct(PackageName(pk,fk), SubscriptionID(pk,fk), ProductID(pk,fk))

SubscriptionPackage(PackageName(pk,fk), SubscriptionID(pk,fk), ProductID(pk,fk))

SignUpFor(VisitorID(pk,fk), ExperienceID(pk,fk))

IngredientAllergens(IngredientID(pk,fk), Allergen(pk))

EmployeeQualifications(EmployeeID(pk,fk), Qualification(pk))

Reservation(ReservationID(pk), EmployeeID(fk), ExperienceID(fk),

AppointmentZoomID(fk), ReservationDate, ReservationExpDate, ReservationTime,

ReservationNumPeople, ReservationDietaryPref)