## 1. Ontwerp het bijbehorende ERD

## A black background with white text AI-generated content may be incorrect.

## 2. Ontwerp de tabellen met behulp van SQL

## Vul de tabellen zodanig met gegevens dat de juistheid van de selecties makkelijk te controleren is.

INSERT INTO Land VALUES ('FR','Frankrijk'), ('ES','Spanje');  
  
INSERT INTO Begeleider VALUES  
 ('Mr. A. Philippo','24, Rue Strassbourg','09-412-6752341','Lyon'),  
 ('Ms. L. Torres','Calle Mayor 18','0034-91-5551234','Madrid');  
  
INSERT INTO Student VALUES  
 ('G5467893','Jan','Slootweg','1987-12-13','HLO'),  
 ('X321632P','Viola','Parmelis','1988-03-24','MEAO'),  
 ('B1122334','Ramon','de Wit','2005-01-11','Economie');  
  
INSERT INTO Vakantie VALUES  
 ('F125','Mr. A. Philippo','FR','Franse Alpen','2004-07-12',10,487),  
 ('E210','Ms. L. Torres', 'ES','Costa Brava','2004-08-05',14,550);  
  
INSERT INTO Boeking VALUES  
 ('F125','G5467893',0.10\*487),  
 ('F125','X321632P',0.10\*487),  
 ('E210','B1122334',0.05\*550);

## Vakanties langer dan 10 dagen

|  |
| --- |
| -- Selecteer alle reizen die meer dan tien dagen duren SELECT VakantieCode,  Bestemming,  AantalDagen,  Prijs FROM Vakantie WHERE AantalDagen > 10; |

## Studenten ouder dan 18 binnen opleiding Economie

|  |
| --- |
| -- Studenten van de opleiding Economie die ouder zijn dan 18 jaar SELECT StudentVoornaam,  StudentAchternaam,  Collegekaart,  Geboortedatum FROM Student WHERE Opleidingen = 'Economie'  AND TIMESTAMPDIFF(YEAR, Geboortedatum, CURDATE()) > 18; |

## Alle studenten gesorteerd op opleiding en leeftijd

|  |
| --- |
| -- Overzicht van alle studenten, eerst op opleiding, dan op geboortedatum SELECT \* FROM Student ORDER BY Opleidingen,  Geboortedatum; |

## Prijs per dag van iedere reis

|  |
| --- |
| -- Bereken een tarief per dag voor elke vakantie SELECT VakantieCode,  Bestemming,  ROUND(Prijs / AantalDagen, 2) AS PrijsPerDag FROM Vakantie; |

## Gemiddelde reissom

|  |
| --- |
| -- Gemiddelde prijs over alle beschikbare vakanties SELECT AVG(Prijs) AS GemiddeldeReisPrijs FROM Vakantie; |

## Student, korting en landcode per boeking

|  |
| --- |
| -- Toon per boeking de student, zijn korting en het land van de reis SELECT s.StudentVoornaam,  s.StudentAchternaam,  b.StudentKorting,  v.LandCode FROM Student s JOIN Boeking b ON b.Collegekaart = s.Collegekaart JOIN Vakantie v ON v.VakantieCode = b.VakantieCode; |

## Reizen met maximaal drie deelnemers

|  |
| --- |
| -- Lijst alle reizen waar hooguit drie studenten voor zijn ingeschreven SELECT v.VakantieCode,  v.Bestemming,  COUNT(b.Collegekaart) AS AantalStudenten FROM Vakantie v LEFT JOIN Boeking b ON b.VakantieCode = v.VakantieCode GROUP BY v.VakantieCode, v.Bestemming HAVING COUNT(b.Collegekaart) <= 3; |