

# **DATABASE ON INCOME TAX DEPARTMENT**

## **(TaxTrekker)**

### **Group Members:**

1. Harsh Patel (202101183)
2. Karan Jivanramjiwala (202101189)
3. Shekhar Gupta (202101203)
4. Devang Rathod (202101204)

### **Group Representative:**

Karan Jivanramjiwala (202101189)  
Phone no - 9408873173

## ★ Functional Dependency

### 1. Tax\_payer(Pan\_number,name,Occupation type,DOB,address,Resident\_no,gender)

**key: Pan\_number**

Fd\_min set -

- Pan\_number -> name
- Pan\_number ->Occupation type
- Pan\_number ->DOB
- Pan\_number ->address
- Pan\_number ->Resident\_no
- Pan\_number ->gender

Thus as **Pan\_number** is super key in every above functional dependency so **Tax\_payer** is in **BCNF**.

### 2. Email\_Tax\_payer(email\_id,pan\_number)

**Key: Email\_id**

Fd\_min set -

- Email\_id -> pan\_number

Thus as **Email\_id** is super key in every above functional dependency so **Email\_Tax\_payer** is in **BCNF**.

### 3. Tax\_Payments(Payment\_id,payment\_amount,tax\_year,payment\_mode)

**Key: Payment\_id**

Fd\_min set -

- Payment\_id -> payment\_amount
- Payment\_id ->tax\_year
- Payment\_id -> payment\_mode

Thus as **Payment\_id** is super key in every above functional dependency so **Tax\_payments** is in **BCNF**

#### 4. **Tax\_rate(rate\_id,tax\_type,tax\_year)**

**Key: rate\_id**

Fd\_min set -

- rate\_id -> tax\_type
- rate\_id -> tax\_year

Thus as **Tax\_rate** is super key in every above functional dependency, so **Tax\_rate** is in **BCNF**.

#### 5. **Tax\_refunds(refund\_id,refund\_amount,refund\_data, tax\_year)**

**Key: refund\_id**

Fd\_min set -

- refund\_id -> refund\_amount
- refund\_id -> refund\_data
- refund\_id -> tax\_year

Thus as **refund\_id** is super key in every above functional dependency, **Tax\_refunds** is in **BCNF**.

#### 6. **Tax\_questions(question\_id,category,description,status)**

**Key: question\_id**

Fd\_min set -

- question\_id -> category
- question\_id -> description
- question\_id -> status

Thus as **question\_id** is super key in every above functional dependency, **Tax\_questions** is in **BCNF**.

#### 7. **Tax\_questions\_asked(question\_id,pan\_number)**

**Key : {question\_id,pan\_number}**

Fd\_min set -

As all attributes, this relation is part of key, so this above relation is in **BCNF**.

#### 8. Tax\_Audits(Audit\_id,Audit\_date,Audit\_description,Audit\_result)

**Key: Audit\_id**

Fd\_min set -

- Audit\_id -> Audit\_date
- Audit\_id -> Audit\_description
- Audit\_id -> Audit\_result

Thus as **Audit\_id** is super key in every above functional dependency, **Tax\_Audits** is in **BCNF**.

#### 9. Tax\_prepares(Preparer\_id,name,Email)

**Key : Preparer\_id**

Fd\_min set -

- Preparer\_id -> name
- Preparer\_id -> Email

Thus as **Preparer\_id** is super key in every above functional dependency, so **Tax\_prepares** is in **BCNF**.

#### 10. prepares(preparer\_id,pan\_number)

**Key : {preparer\_id,pan\_number}**

Fd\_min set -

As all attributes of this relation are part of the key, this above relation is in **BCNF**.

#### 11. Income\_Tax\_Branches(Office\_id,HOD, address)

**Key: Office\_id**

Fd\_min set -

- Office\_id -> HOD
- Office\_id -> address

Thus as **Office\_id** is super key in every above functional dependency so **Income\_Tax\_Branches** is in **BCNF**.

## 12. Bank(Bank\_name,account\_no,Account\_type)

**Key : {Bank\_name,account\_no}**

Fd\_min set -

- {Bank\_name,account\_no} -> Account\_type

Thus as {Bank\_name,account\_no} is super key in every above functional dependency, **Bank** is in **BCNF**.

## 13. has\_Account(Bank\_nam,account\_no,Pan\_number)

**Key : {Bank\_nam,account\_no,Pan\_number}**

Fd\_min set -

As all attributes of this relation are part of the key, this above relation is in **BCNF**.

## 14. Assets(Asset\_id,Acquisition\_date,acquisition\_cost, asset\_description)

**Key: Asset\_id**

Fd\_min set -

- Asset\_id -> Acquisition\_date
- Asset\_id -> Acquisition\_cost
- Asset\_id -> asset\_description

Thus as **Asset\_id** is super key in every above functional dependency so **Assets** is in **BCNF**.

## 15. Tax\_documents(Doc\_id,Doc\_type, doc\_date\_generated)

**Key: Doc\_id**

Fd\_min set -

- Doc\_id -> Doc\_type
- Doc\_id -> Doc\_date\_generated

Thus as **Doc\_id** is super key in every above functional dependency so **Tax\_documents** is in **BCNF**.

#### 16. Income(pan\_number,income\_type,income\_amount, income\_date)

**Key : {pan\_number,income\_type}**

Fd\_min set -

- {pan\_number,income\_type} -> income\_amount
- {pan\_number,income\_type} -> income\_date

Thus as{**pan\_number,income\_type**} is super key in every above functional dependency, so **Income** is in **BCNF**.

#### 17. Tax\_credits(credit\_id -> credit\_type, credit\_amount,Tax\_year)

**Key: credit\_id**

Fd\_min set -

- credit\_id -> credit\_type
- credit\_id -> credit\_amount
- credit\_id -> Tax\_year

Thus as **credit\_id** is super key in every above functional dependency, **Tax\_credits** is in **BCNF**.

#### 18. Tax\_withholdings(Withholding\_id,employer\_name,tax\_year, income\_percentage)

**Key: Withholding\_id**

Fd\_min set -

- Withholding\_id -> employer\_name
- Withholding\_id -> tax\_year
- Withholding\_id -> income\_percentage

Thus as **Withholding\_id** is super key in every above functional dependency, **Tax\_withholdings** is in **BCNF**.

## 19. Tax\_forms(form\_id,form\_type,form\_status,form\_due\_date)

**Key: form\_id**

Fd\_min set -

- form\_id -> form\_type
- form\_id -> form\_status
- form\_id -> form\_due\_date

Thus as **form\_id** is super key in every above functional dependency, **Tax\_forms** is in **BCNF**.

## 20. Tax\_dependents(pan\_number,Dependent\_name,relation,DOB,gender)

**Key : {pan\_number,Dependent\_name}**

Fd\_min set -

- {pan\_number,Dependent\_name} -> relation
- {pan\_number,Dependent\_name} -> DOB
- {pan\_number,Dependent\_name} -> gender

Thus as **{pan\_number, Dependent\_name}** is super key in every above functional dependency, **Tax\_dependents** is in **BCNF**.

### ★ DDL script

CREATE TABLE Assets

```
(
  Asset_id      int NOT NULL,
  Pan_number    bigint NOT NULL,
  Acquisition_date date NOT NULL,
  Acquisition_cost bigint NOT NULL,
  Asset_description varchar(50) NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( Asset_id ),
  CONSTRAINT FK_6 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer ( Pan_number
);
```

```
CREATE INDEX FK_2 ON Assets
(
  Pan_number
);
```

```
CREATE TABLE Bank
(
  Bank_name varchar(50) NOT NULL,
  Account_no bigint NOT NULL,
  Account_type varchar(50) NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( Bank_name, Account_no )
);
```

```
CREATE TABLE Email_Tax_payer
(
  Email_id varchar(50) NOT NULL,
  Pan_number bigint NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( Email_id ),
  CONSTRAINT FK_17 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```

```
CREATE INDEX FK_2 ON Email_Tax_payer
(
  Pan_number
);
```

```
CREATE TABLE has_Account
(
  Bank_name varchar(50) NOT NULL,
  Account_no bigint NOT NULL,
  Pan_number bigint NOT NULL,
  CONSTRAINT PK_2 PRIMARY KEY ( Bank_name, Account_no, Pan_number ),
  CONSTRAINT FK_12 FOREIGN KEY ( Bank_name, Account_no ) REFERENCES Bank (
Bank_name, Account_no ),
  CONSTRAINT FK_23_1 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```



```
CREATE INDEX FK_1 ON has_Account  
(  
    Bank_name,  
    Account_no  
);
```

```
CREATE INDEX FK_3 ON has_Account  
(  
    Pan_number  
);
```

```
CREATE TABLE Income  
(  
    Income_type varchar(50) NOT NULL,  
    Pan_number bigint NOT NULL,  
    income_amount bigint NOT NULL,  
    income_date date NOT NULL,  
    CONSTRAINT PK_1 PRIMARY KEY ( Income_type, Pan_number ),  
    CONSTRAINT FK_18 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer ( Pan_number )  
);
```

```
CREATE INDEX FK_2 ON Income  
(  
    Pan_number  
);
```

```
CREATE TABLE Income_Tax_Branches  
(  
    Office_id int NOT NULL,  
    HOD varchar(50) NOT NULL,  
    pincode bigint NOT NULL,  
    city varchar(50) NOT NULL,  
    "state" varchar(50) NOT NULL,  
    CONSTRAINT PK_1 PRIMARY KEY ( Office_id )  
);
```

```
CREATE TABLE prepares
(
    preparer_id int NOT NULL,
    Pan_number bigint NOT NULL,
    CONSTRAINT PK_2 PRIMARY KEY ( preparer_id, Pan_number ),
    CONSTRAINT FK_9 FOREIGN KEY ( preparer_id ) REFERENCES Tax_preparers (
preparer_id ),
    CONSTRAINT FK_10 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```

```
CREATE INDEX FK_1 ON prepares
(
    preparer_id
);
```

```
CREATE INDEX FK_3 ON prepares
(
    Pan_number
);
```

```
CREATE TABLE Tax_Audits
(
    Audit_id      int NOT NULL,
    Pan_number    bigint NOT NULL,
    Audit_date     date NOT NULL,
    Audit_description varchar(50) NOT NULL,
    Audit_result   varchar(50) NOT NULL,
    CONSTRAINT PK_1 PRIMARY KEY ( Audit_id ),
    CONSTRAINT FK_11 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```

```
CREATE INDEX FK_2 ON Tax_Audits
(
  Pan_number
);
```

```
CREATE TABLE Tax_credits
(
  credit_id    int NOT NULL,
  credit_type  varchar(50) NOT NULL,
  Pan_number   bigint NOT NULL,
  credit_amount bigint NOT NULL,
  Tax_year     int NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( credit_id ),
  CONSTRAINT FK_22_1 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```

```
CREATE INDEX FK_2 ON Tax_credits
(
  Pan_number
);
```

```
CREATE TABLE Tax_dependents
(
  Dependent_name varchar(50) NOT NULL,
  Pan_number      bigint NOT NULL,
  relation        varchar(50) NOT NULL,
  DOB             date NOT NULL,
  gender          varchar(50) NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( Dependent_name, Pan_number ),
  CONSTRAINT FK_16 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```

```
CREATE INDEX FK_2 ON Tax_dependents
(
  Pan_number
);
```

```
CREATE TABLE Tax_documents
(
  Doc_id          bigint NOT NULL,
  Pan_number      bigint NOT NULL,
  Doc_type        varchar(50) NOT NULL,
  Doc_date_generated date NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( Doc_id ),
  CONSTRAINT FK_19 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```

```
CREATE INDEX FK_2 ON Tax_documents
(
  Pan_number
);
```

```
CREATE TABLE Tax_forms
(
  form_id        bigint NOT NULL,
  Pan_number     bigint NOT NULL,
  form_type      varchar(50) NOT NULL,
  form_status    varchar(50) NOT NULL,
  form_due_date  date NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( form_id ),
  CONSTRAINT FK_22 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
Pan_number )
);
```

```
CREATE INDEX FK_2 ON Tax_forms
(
  Pan_number
);
```

```
CREATE TABLE Tax_payer
(
  Pan_number    bigint NOT NULL,
  "First name"  varchar(50) NOT NULL,
  Office_id     int NOT NULL,
  "Last name"   varchar(50) NOT NULL,
  "Occupation Type" varchar(50) NOT NULL,
  DOB          date NOT NULL,
  pincode       varchar(50) NOT NULL,
  city          varchar(50) NOT NULL,
  "state"       varchar(50) NOT NULL,
  Resident_no   int NOT NULL,
  gender        varchar NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( Pan_number ),
  CONSTRAINT FK_22_2 FOREIGN KEY ( Office_id ) REFERENCES Income_Tax_Branches (
Office_id )
);
```

```
CREATE INDEX FK_2 ON Tax_payer
(
  Office_id
);
```

```
CREATE TABLE Tax_Payments
(
  payment_id    int NOT NULL,
  pan_number    bigint NOT NULL,
  Payment_Amount bigint NOT NULL,
  Tax_year      int NULL,
  Payment_mode  varchar(50) NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( payment_id ),
  CONSTRAINT FK_3 FOREIGN KEY ( pan_number ) REFERENCES Tax_payer ( Pan_number
)
);
```

```
CREATE INDEX FK_2 ON Tax_Payments
(
    pan_number
);
```

```
CREATE TABLE Tax_preparers
(
    preparer_id int NOT NULL,
    Office_id int NOT NULL,
    name varchar(50) NOT NULL,
    email varchar NOT NULL,
    CONSTRAINT PK_1 PRIMARY KEY ( preparer_id ),
    CONSTRAINT FK_8 FOREIGN KEY ( Office_id ) REFERENCES Income_Tax_Branches (
Office_id )
);
```

```
CREATE INDEX FK_2 ON Tax_preparers
(
    Office_id
);
```

```
CREATE TABLE Tax_questions
(
    question_id int NOT NULL,
    category varchar(50) NOT NULL,
    description varchar(50) NOT NULL,
    status varchar(50) NOT NULL,
    CONSTRAINT PK_1 PRIMARY KEY ( question_id )
);
```

```
CREATE TABLE Tax_questions_asked
(
    question_id int NOT NULL,
    Pan_number bigint NOT NULL,
    CONSTRAINT PK_3 PRIMARY KEY ( question_id ),
    CONSTRAINT FK_1 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer ( Pan_number
),
    CONSTRAINT FK_2 FOREIGN KEY ( question_id ) REFERENCES Tax_questions (
question_id )
);
```

```
CREATE INDEX FK_1 ON Tax_questions_asked
(
  Pan_number
);
```

```
CREATE INDEX FK_2 ON Tax_questions_asked
(
  question_id
);
```

```
CREATE TABLE Tax_rate
(
  rate_id int NOT NULL,
  Pan_number bigint NOT NULL,
  Tax_type varchar(50) NOT NULL,
  Tax_year int NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( rate_id ),
  CONSTRAINT FK_5 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer ( Pan_number
)
);
```

```
CREATE INDEX FK_2 ON Tax_rate
(
  Pan_number
);
```

```
CREATE TABLE Tax_refunds
(
  refund_id int NOT NULL,
  Pan_number bigint NOT NULL,
  refund_amount bigint NOT NULL,
  refund_date date NOT NULL,
  Tax_year int NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( refund_id ),
  CONSTRAINT FK_7 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer ( Pan_number
)
);
```

```
CREATE INDEX FK_2 ON Tax_refunds
(
  Pan_number
);
```

```
CREATE TABLE Tax_withholdings
(
  withholding_id  int NOT NULL,
  Pan_number      bigint NOT NULL,
  Tax_year        int NOT NULL,
  employer_name   varchar(50) NOT NULL,
  Income_percentage int NOT NULL,
  CONSTRAINT PK_1 PRIMARY KEY ( withholding_id ),
  CONSTRAINT FK_23 FOREIGN KEY ( Pan_number ) REFERENCES Tax_payer (
    Pan_number )
);
```

```
CREATE INDEX FK_2 ON Tax_withholdings
(
  Pan_number
);
```

### **Some changes that we have made in ER diagram:-**

- 1) We have removed some relationships, such as between Tax\_Audits and Tax\_documents, Tax\_withholdings, and Income,Tax\_forms and Income\_tax\_department,Tax\_Documnets and Income\_tax\_department.
- 2) We have removed some attributes from Income\_tax\_department, Tax\_payer, and Tax\_withholdings.
- 3) We have changed some cardinality constraints in some relationships such as posses(between Tax\_payer and assets), Audited to( Tax\_payer and Tax\_audits) and eligible for(Tax\_payer and Tax\_credits).



