

```

CREATE TABLE ROLE (
    ROLE VARCHAR(255) NOT NULL,
    Description VARCHAR(50),
    Wage DECIMAL(10, 2),
    PRIMARY KEY (ROLE)
);

CREATE TABLE WAGE (
    EMPLOYEE_ID INT NOT NULL,
    HOURLY DECIMAL(10, 2),
    SALARY DECIMAL(10, 2),
    PRIMARY KEY (EMPLOYEE_ID)
);

CREATE TABLE SALARY (
    EMPLOYEE_ID INT NOT NULL,
    SALARY DECIMAL(10, 2),
    ROLE VARCHAR(255),
    PRIMARY KEY (EMPLOYEE_ID),
    FOREIGN KEY (EMPLOYEE_ID) REFERENCES WAGE(EMPLOYEE_ID) ON DELETE
    CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (ROLE) REFERENCES ROLE(ROLE) ON DELETE CASCADE ON
    UPDATE CASCADE
);

CREATE TABLE SHIFT (
    SHIFT_ID INT NOT NULL,
    SHIFT_NAME VARCHAR(255),
    START_TIME TIME,
    END_TIME TIME,
    SHIFT_TYPE VARCHAR(255),
    PRIMARY KEY (SHIFT_ID)
);

CREATE TABLE CUSTOMER (
    CUSTOMER_ID INT NOT NULL,
    NAME VARCHAR(255),
    ADDRESS VARCHAR(255),
    PHONE VARCHAR(20),
    POINTS_BALANCE INT,
    PRIMARY KEY (CUSTOMER_ID)
);

CREATE TABLE VENDOR (
    VENDOR_ID INT NOT NULL,
    NAME VARCHAR(255),
    PHONE VARCHAR(20),
    SUPPLY_TYPE VARCHAR(255),
    PRIMARY KEY (VENDOR_ID)
);

```

```
CREATE TABLE ITEM (  
    ITEM_ID INT NOT NULL,  
    ITEM_NAME VARCHAR(255),  
    CATEGORY VARCHAR(255),  
    PRICE DECIMAL(10, 2),  
    SIZE VARCHAR(255),  
    HISTORY_P TEXT,  
    PRIMARY KEY (ITEM_ID)  
);
```

```
CREATE TABLE ORDER_STATUS (  
    STATUS_ID INT NOT NULL,  
    STATUS VARCHAR(255),  
    TIMESTAMP DATETIME,  
    PRIMARY KEY (STATUS_ID)  
);
```

```
CREATE TABLE EMPLOYEE (  
    EMPLOYEE_ID INT NOT NULL,  
    NAME VARCHAR(255),  
    ADDRESS VARCHAR(255),  
    DOB DATE,  
    HIRE_DATE DATE,  
    ROLE_ID VARCHAR(255),  
    WAGE_ID INT,  
    PRIMARY KEY (EMPLOYEE_ID),  
    FOREIGN KEY (ROLE_ID) REFERENCES ROLE(ROLE) ON DELETE CASCADE ON  
UPDATE CASCADE,  
    FOREIGN KEY (WAGE_ID) REFERENCES WAGE(EMPLOYEE_ID) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE SUPERVISOR (  
    SUPER_ID INT NOT NULL,  
    EMPLOYEE_ID INT,  
    PRIMARY KEY (SUPER_ID),  
    FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON  
DELETE CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE CLOCK_IN (  
    CLOCK_IN_ID INT NOT NULL,  
    EMPLOYEE_ID INT,  
    SHIFT_ID INT,  
    CLOCK_IN_TIME DATETIME,  
    CLOCK_OUT_TIME DATETIME,  
    PRIMARY KEY (CLOCK_IN_ID),  
    FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON
```

```
DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (SHIFT_ID) REFERENCES SHIFT(SHIFT_ID) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE CLUB_CARD (  
    CARD_ID INT NOT NULL,  
    CUSTOMER_ID INT,  
    POINTS INT,  
    PRIMARY KEY (CARD_ID),  
    FOREIGN KEY (CUSTOMER_ID) REFERENCES CUSTOMER(CUSTOMER_ID) ON  
DELETE CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE CERTIFICATION (  
    CERT_ID INT NOT NULL,  
    EMPLOYEE_ID INT,  
    CERTIFICATION VARCHAR(255),  
    EXPIRE DATE,  
    PRIMARY KEY (CERT_ID),  
    FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON  
DELETE CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE JOB_HISTORY (  
    JOB_H_ID INT NOT NULL,  
    EMPLOYEE_ID INT,  
    PREVIOUS_JOB VARCHAR(255),  
    START_DATE DATE,  
    END_DATE DATE,  
    SUPER_ID INT,  
    PRIMARY KEY (JOB_H_ID),  
    FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID),  
    FOREIGN KEY (SUPER_ID) REFERENCES SUPERVISOR(SUPER_ID)  
);
```

```
CREATE TABLE [ORDER] (  
    ORDER_ID INT NOT NULL,  
    CUSTOMER_ID INT,  
    ORDER_DATE DATE,  
    TOTAL DECIMAL(10, 2),  
    RATING INT,  
    PRIMARY KEY (ORDER_ID),  
    FOREIGN KEY (CUSTOMER_ID) REFERENCES CUSTOMER(CUSTOMER_ID) ON  
DELETE CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE SALARY_CERTIFICATION (  
    EMPLOYEE_ID INT,
```

```
    CERT_ID INT,  
    DATE DATE,  
    PRIMARY KEY (EMPLOYEE_ID, CERT_ID),  
    FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON  
DELETE NO ACTION ON UPDATE CASCADE,  
    FOREIGN KEY (CERT_ID) REFERENCES CERTIFICATION(CERT_ID) ON DELETE  
CASCADE ON UPDATE NO ACTION  
);
```

```
CREATE TABLE INVENTORY_ITEM (  
    INVENTORY_ITEM_ID INT NOT NULL,  
    ITEM_NAME VARCHAR(255),  
    IN_STOCK INT,  
    REORDER_LEVEL INT,  
    VENDOR_ID INT,  
    PRIMARY KEY (INVENTORY_ITEM_ID),  
    FOREIGN KEY (VENDOR_ID) REFERENCES VENDOR(VENDOR_ID) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE ORDER_ITEM (  
    ORDER_ITEM_ID INT NOT NULL,  
    ORDER_ID INT,  
    ITEM_ID INT,  
    QUANTITY INT,  
    UNIT_PRICE DECIMAL(10, 2),  
    HISTORY_P TEXT,  
    PRIMARY KEY (ORDER_ITEM_ID),  
    FOREIGN KEY (ORDER_ID) REFERENCES [ORDER](ORDER_ID) ON DELETE  
CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (ITEM_ID) REFERENCES ITEM(ITEM_ID) ON DELETE CASCADE  
ON UPDATE CASCADE  
);
```

```
CREATE TABLE PAYMENT (  
    PAY_ID INT NOT NULL,  
    ORDER_ID INT,  
    PAYMENT DECIMAL(10, 2),  
    AMOUNT DECIMAL(10, 2),  
    PRIMARY KEY (PAY_ID),  
    FOREIGN KEY (ORDER_ID) REFERENCES [ORDER](ORDER_ID) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```