

# Physical design and implementation essay examples

[Law](#), [Security](#)



## **The professor's name**

### Database security

Database security is where organizational resources are secured against unauthorized reading, or alterations. Databases are at risk of unauthorized access through different channels especially those connected to a network.

Database security can be categorized into two; system security and data security. The system security in this perspective is not security of the operating system but internal database operating system. A database system like Oracle system has its own operating system. Integrity is an example of data security. Securing DBMS system involves controlling access to the system and how the database system level.

Databases are secured by ensuring that combination of valid username/password. The amount of disk space available for the user and limitation of resources to users is also security mechanism. The mechanism for database system security by ensuring that only authorized users are connected to the database. The system audit must be activated to ensure that all unauthorized access is denied. Security to data is enforced by also ensuring there are control access and usage of the database system. Users are given access to schema object and are also allowed actions to do on the systems. Actions that users can be allowed to perform include; SELECT, INSERT, DELETE and UPDATE.

Access controls are enforced to both individual users and the general public which are defined by Oracle databases. Access to data is enforced by giving permissions on tables, views and database operations to a user. Permission can also be granted to database objects such as the forms, reports and

graphs. Oracle database system has the following security features to control access and how it is used. The security mechanism ensures that unauthorized database access, prevent unauthorized access to schema objects and control disk usage. Security control is overly taken care of by the user account, the privileges, role and profiles.

## **How to defeat SQL injection attacks**

SQL injection attacks occur to internet based database systems where attackers try to modify SQL commands by exploiting improper SQL statements. The attackers are aimed at accessing the system as an administrator without valid username and password. They are also aimed at accessing sensitive information from the system in a direct or indirect manner. Attackers can also impact on integrity of data in the database using the SQL injection attackers. The attack can be defeated at two levels; in the application and network. The application system where the injection is done can be used to defeat the attacker.

## **Physical database design**

This is the process of implementing logical database design in database management systems to a structural model that consists of records, files and other physical structures. Physical database design involves creating relations. This stage in database design is highly critical since the speed of access to database depends on it. Accessibility, database security, and user friendliness are also significant issues to be considered under this section. There are several challenges that face the process of physical design. Size of the database, where concerns are; number of relations, number of tables

and the size of each table. Database usage such as delete, update, insert and queries is also another challenge of physical design.

## SQL Code

Creating a database

```
CREATE DATABASE `Campix`;
```

Creating a table for users

```
CREATE TABLE `user` (  
  `UserName` VARCHAR( 20 ) NOT NULL , `Password` VARCHAR( 20 ) NOT  
  NULL , `Login_time` TIMESTAMP NOT NULL , `UserId` TINYINT( 3 ) NOT NULL ,  
  PRIMARY KEY ( `UserName` )  
) ENGINE = innodb;
```

## CREATE TABLE `Company` (

```
`CompanyID` VARCHAR( 12 ) NOT NULL , `CompanyName` VARCHAR( 20 )  
NOT NULL , `CompaniesHouseNumber` VARCHAR( 255 ) NOT NULL , `SEDOL`  
VARCHAR( 255 ) NOT NULL , `Address1` VARCHAR( 50 ) NOT NULL , `City`  
VARCHAR( 50 ) NOT NULL , `Country` VARCHAR( 255 ) NOT NULL , `Email`  
VARCHAR( 50 ) NOT NULL , `Website` VARCHAR( 255 ) NOT NULL , PRIMARY  
KEY ( `CompanyID` )  
) ENGINE = innodb;
```

## CREATE TABLE `CompanyRelationships` (

```
`RelationshipID` INT( 25 ) NOT NULL , `EarlierCompanyID` VARCHAR( 255 )  
NOT NULL , `LaterCompanyID` VARCHAR( 255 ) NOT  
NULL , `NatureOfRelationship` VARCHAR( 255 ) NOT NULL , `EditName`
```

```
VARCHAR( 255 ) NOT NULL , `EditInst` VARCHAR( 255 ) NOT NULL , `EditDate`  
DATE NOT NULL , PRIMARY KEY ( `RelationshipID` )  
) ENGINE = innodb;
```

### **CREATE TABLE `DocumentSeries` (**

```
`DocumentSeriesID` INT( 5 ) NOT NULL , `CompanyID` INT( 255 ) NOT  
NULL , `DocumentTypeID` VARCHAR( 255 ) NOT NULL , `SeriesName`  
VARCHAR( 255 ) NOT NULL , `EditDate` DATE NOT NULL , PRIMARY KEY  
( `DocumentSeriesID` )  
) ENGINE = innodb;
```

### **CREATE TABLE `DocumentSeriesCollection` (**

```
`DSCID` INT( 12 ) NOT NULL , `CompanyID` INT( 12 ) NOT  
NULL , `DocumentSeriesID` INT( 12 ) NOT NULL , `DocumentTypeID` INT( 12 )  
NOT NULL , `CollectionID` INT( 12 ) NOT NULL , `StartingDate` DATE NOT  
NULL , `EndDate` DATE NOT NULL , `SeriesName` VARCHAR( 255 ) NOT  
NULL , PRIMARY KEY ( `DSCID` )  
) ENGINE = innodb;
```

### **CREATE TABLE `DocumentHoldingsn` (**

```
`HoldingID` VARCHAR( 12 ) NOT NULL , `CompanyID` VARCHAR( 12 ) NOT  
NULL , `CollectionID` VARCHAR( 12 ) NOT NULL , `DocumentSeriesID`  
VARCHAR( 12 ) NOT NULL , `IndividualYear` DATE NOT NULL , `Location`  
VARCHAR( 255 ) NOT NULL , `DateReceived` DATE NOT NULL  
) ENGINE = innodb;
```

## References

" Oracle DBA Handbook" 7. 3 Edition, by Kevin Loney; McGraw Hill; ISBN 0-07-882289-0.

" Oracle Database Backup" by Curtis Preston; July, 1996 Sys Admin. Magazine.

Simon, A. R. (1999). Strategic Database Technology: Management for the Year 2000, San Francisco, CA: Morgan Kaufmann Publishers.

Theriault, M and Newman, A. (2001). Oracle Security Handbook, Osborne/McGraw-Hill.

" Using Database Functions in SQL Injection Attacks" retrieved from <http://www.integrigy.com/security-resources>

Appendix

Screen shots of the table