user avatar

Ophir Frieder

00:03

Good morning.

00:05

Good afternoon, good evening, it clearly depends on the time that you're watching this but for me it is 7pm on Wednesday night on march 16.

00:20

By now, you all have seen the demos, of the covert project as you've seen the four groups that presented they each had a diversity of different approaches different data.

00:34

But they all find out that the data speaks for itself, it speaks for what is actually there and not necessarily for what the cause for why it's there.

00:46

Today, just to cap, where we are.

00:50

We are in a mode that has seen us.

00:56

Let me share the screen.

01:00

we're in a mode, that is, had us show where we are in terms of the class, as you can tell.

01:10

we've covered most of the class, thus far, at least as far as concerned, for what you need to implement, we had an introduction.

01:21

We had an introduction to the relational model and describe what the relational model is and what it's not we talked about the history that predated it and the motivation for it.

01:34

We then had a discussion on the relational algebra the foundations for the relational model.

01:41

We concluded with a commentary on the closeness of the model, so that all operations yield relations, and then we talked about the structured query language sql, which is the programming languages for relational databases.

02:00

We described how to design and then we've covered just recently, the query processing and optimization techniques.

02:10

What remains in the course is the transaction concurrency recovery integration of structured data and text big data and data warehousing mine and we will cover all that.

02:24

However, none of that remaining material is related to what you need to do in your projects.

02:30

And the reason being is we are assuming a system that doesn't crash, so you don't need to worry about recovery.

早上好。

00:05

下午好，晚上好，这显然取决于你看这个的时间，但对我来说是 3 月 16 日星期三晚上 7 点。

00:20

到目前为止，你们都已经看到了秘密项目的演示，因为你们已经看到了四个小组，他们每个小组都有不同的方法，不同的数据。

00:34

但他们都发现数据不言自明，它说明了实际存在的东西，而不一定说明它存在的原因。

00:46

今天，只是为了上限，我们在哪里。

00:50

我们处于一种已经看到我们的模式。

00:56

让我分享屏幕。

01:00

正如你所知道的，我们处于一种模式，也就是说，让我们展示我们在班级方面的位置。

01:10

我们已经涵盖了大部分课程，到目前为止，至少就您需要实现的内容而言，我们有一个介绍。

01:21

我们对关系模型进行了介绍，并描述了关系模型是什么，不是什么，我们讨论了它之前的历史和它的动机。

01:34

然后我们讨论了关系代数是关系模型的基础。

01:41

最后，我们对模型的接近性进行了评论，以便所有操作都产生关系，然后我们讨论了结构化查询语言 sql，它是关系数据库的编程语言。

02:00

我们描述了如何设计，然后我们刚刚介绍了查询处理和优化技术。

02:10

课程中剩下的就是结构化数据和文本大数据和数据仓库的事务并发恢复集成，我们将涵盖所有这些。

02:24

但是，剩下的所有材料都与您在项目中需要做的事情无关。

02:30

原因是我们假设系统不会崩溃，因此您无需担心恢复问题。

02:37

We assume a single user system, so we don't have to really worry about concurrency control and the transactions is for the sense of making sure that you can have an integrated inner eruption without causing any problem and again we're a single user.

02:53

So basically what it takes for your projects you've covered already.

02:59

The first assignment, as you know, was on data ethics, this was in order to make sure that you know how to use the data in a reliable and honest broker manner with ethical considerations.

03:12

Having covered a little further project to, as you just completed talked about the structured query language in design application it tested those two units by having you develop a application for coven processing.

03:28

We are now moving towards actually understanding the inner workings of a relational database management system, namely, how is such a system developed.

03:38

And what you're going to do is you're going to develop a system that's going to quote take the squat and structured query language.

03:46

And allow you to create a design, by defining a set of tables on a relational schema and hopefully processed the requests and the sql statements.

03:59

in such a way that you can do it in a clean, efficient and optimized manner, so all those operations that we've described, up to now you're going to rely to harness this following system.

04:12

So what is this system you're going to do, let me begin by showing you.

04:18

What your project is going to involve.

04:22

Your project is going to involve the following.

04:27

you're going, this is your assignment.

04:33

And I will make sure that i'm sharing the screen.

04:46

This is your.

04:48

Project three.

04:51

And i'll read it slowly, and then we will discuss it in greater detail, as you know by now, we have learned the ethical data practices, I was project one.

05:03

we've also learned how to design and implement relational database applications mainly your coven processing that was project to it is now time to learn how to implement the underlying structures and methods of relational database management.

05:18

As such, to successfully complete this project, you will design document and implement an actual yet simplified single user relational database management system.

05:32

Your system.

05:34

must include in sql parse or, namely to be able to parse your components.

05:41

And indexing structure.

05:44

Namely, you must support indexes.

05:48

and execution engine Emily you must actually execute and optionally but yet highly recommended a query optimizer at least to some reasonable level.

05:59

As before, you can work in groups of two or three but, unlike the other projects each member is going to assess each other.

06:09

And, as well as the project itself in the final grading So what will happen is the group will get a grade, whatever the grade score may be i'd say 90.

06:22

And they'll be a plus or minus to that score, based on the individual ratings and contribution of each member in the group and you yourselves will give me a guidance of what that is.

06:34

You will demonstrate your project online in 30 minutes as you did, with a covert structure, except now you have 30 minutes.

06:44

I will leave the flexibility of 45 minutes for questioning and answering so the total of 45 minutes, as always, you will send me an advance your write ups and your documentation and I will look at it and we will schedule a time to see what you've done what you've not done.

07:04

Should you report.

07:06

And item being a not complete, I will Doc less than if you do not disclose it and I catch it during the DEMO unlike your other approaches this DEMO is pretty straightforward to see what was done, it was data ethics was fairly easy to deal with.

07:26

In terms of coven most people that most things done right in terms of this application, it will really separate the groups between the ones that have done something and the ones that wish they had done something.

07:42

02:37

我们假设一个单用户系统，所以我们不必真正担心并发控制，事务是为了确保您可以拥有一个集成的内部爆发而不会造成任何问题，并且我们再次是单用户。

02:53

所以基本上你已经涵盖的项目需要什么。

02:59

如您所知，第一项任务是关于数据道德，这是为了确保您知道如何以可靠和诚实的经纪人方式使用数据，并考虑到道德问题。

03:12

在介绍了一个更进一步的项目之后，当您刚刚完成讨论了设计应用程序中的结构化查询语言时，它通过让您开发一个用于 coven 处理的应用程序来测试这两个单元。

03:28

我们现在正朝着真正理解关系数据库管理系统的内部工作原理迈进，即这样的系统是如何开发的。

03:38

你要做的是开发一个系统，该系统将引用深蹲和结构化查询语言。

03:46

并允许您通过在关系模式上定义一组表并希望处理请求和 sql 语句来创建设计。

03:59

以这样一种方式，您可以以一种干净、高效和优化的方式来完成它，所以我们描述的所有这些操作，到目前为止，您将依赖于利用以下系统。

04:12

那么你要做什么这个系统，让我先向你展示。

04:18

你的项目将涉及什么。

04:22

您的项目将涉及以下内容。

04:27

你去，这是你的任务。

04:33

我会确保我正在共享屏幕。

04:46

这是你的。

04:48

项目三。

04:51

我会慢慢阅读它，然后我们将更详细地讨论它，正如你现在所知道的，我们已经学习了道德数据实践，我是项目一。

05:03

我们还学习了如何设计和实现关系数据库应用程序，主要是您的项目处理，现在是学习如何实现关系数据库管理的底层结构和方法的时候了。

05:18

因此，为了成功完成这个项目，您将设计文档并实现一个实际但简化的单用户关系数据库管理系统。

05:32

你的系统。

05:34

必须包含在 sql 解析中，或者，即能够解析您的组件。

05:41

和索引结构。

05:44

即，您必须支持索引。

05:48

和执行引擎 Emily，您必须实际执行，并且可以选择但强烈推荐查询优化器，至少达到某种合理的水平。

05:59

和以前一样，您可以以两三个人为一组工作，但与其他项目不同，每个成员都将相互评估。

06:09

而且，以及最终评分中的项目本身所以会发生的是该小组将获得一个成绩，无论成绩是多少，我都会说 90。

06:22

根据小组中每个成员的个人评分和贡献，他们将对该分数加分或减分，你们自己会给我一个指导。

06:34

您将像以前一样在 30 分钟内在线演示您的项目，使用隐蔽的结构，但现在您有 30 分钟。

06:44

我将留出 45 分钟的灵活时间进行提问和回答，因此总共 45 分钟，一如既往，您将提前向我发送您的文章和文档，我会查看它，我们会安排时间来看看您的内容'已经做了你没有做过的事。

07:04

该不该报。

07:06

并且项目不完整，如果您不披露它，我将记录更少，并且我在演示期间捕获它与您的其他方法不同，此演示非常简单地看到做了什么，这是数据伦理很容易处理.

07:26

对于大多数人来说，大多数事情在这个应用程序方面都做得很好，它确实将已经做了某事的人和希望他们做了某事的人分开了。

07:42

The good news is you're two thirds done with the projects The bad news is you've got about a little more than half the grade pending on the remainder of the course so.

07:54

From now on, this is your focus, it will be do near the end of the Semester and it will be done in demos on the last day of class in May early May.

08:06

We will have the best teams present as we did this time for the covert applications and you will see a wide variance of approaches and solutions along the way.

08:17

So how are you going to do this and what's the guidelines of how to do this well, let me.

08:25

show you the proper way so i'll stop sharing.

08:31

I will switch documents.

08:36

And I will share once again.

user avatar

未知演讲者

08:45

Let me.

user avatar

Ophir Frieder

08:47

show that again.

08:58

So this is your cheat sheet or guidelines and again this is purely a gentle guidelines or suggestions feel free.

09:08

To do it completely differently any there are no absolute one right way of doing things, some are going to be easier so i'm going to be harder.

09:21

You should use the programming language you feel most comfortable in you should use the style of cooperation, the way you most feel comfortable in.

09:30

So.

09:32

How should we do this.

09:34

Well, let me begin by telling you what way I would design, this is based on not only many years of teaching, but many different group feedback from many different systems that were developed so here are some of the hints of how to go about it.

09:54

Basically, your architecture is straightforward, it has six main components, it has a query input manager.

10:04

A data definition language, a data manipulation language some way of executing it in this case it's a main memory system and a storage structure in this case it is a disk and with all that you need to form some level of optimization.

10:23

So i'm going to describe each one of these in details, as I mentioned, this is not a must do this way, this is just an easier way of approaching it and again many solutions are right.

10:38

So query input manager.

10:43

What I saw what i've seen over the years that people spend a lot of time and it's not the goal of this class is a beautiful query input manager, they have a beautiful interface, they have a nice way of displaying results that's all fine.

11:01

But that's not the goal of this class a simple line editor or simple visual editor is more than enough because our goal is to show and understand the relational database management system, how it operates not how the user interface operates now, if you wish to do so by.

11:24

In order to simplify your parsing.

11:27

that's Okay, but your goal should be to make sure that you can parse the solution, you can parse the query as it is given, namely the sql query that is written as you've done so in project number two.

11:45

Would you must do is, you must find the sql grammar.

11:49

recording the grammar is pointless and very error error error prone.

11:55

You can find it online, you can find it any which way you want, and therefore you can rely as any and use any kind of poor posture that you wish the simplest parts are is a recursive descent par Sir.

12:11

You should take.

12:14

You should use a recursive disempower Sir because it's easier, but you don't have to.

12:20

If you find a partner that you wish to use, you should state the source that you're using that is not necessarily to find.

12:29

A system that does it for you, but if you way see us a parse or that takes and parses sql language again, this is not the point of this course, I am not teaching you parsing and hence that's why state that recursive disempower Sir is more than enough but.

好消息是你已经完成了三分之二的项目 坏消息是你在剩下的课程中还有大约一半的成绩需要等待。

07:54

从现在开始，这是你的重点，它将在学期末完成，并将在 5 月初的最后一天课程的演示中完成。

08:06

我们将拥有最好的团队，就像我们这次为秘密应用程序所做的那样，您将在此过程中看到各种各样的方法和解决方案。

08:17

那么你将如何做到这一点，以及如何做好这件事的指导方针是什么，让我来看看。

08:25

告诉你正确的方法，所以我会停止分享。

08:31

我会切换文件。

08:36

我将再次分享。

用户头像

未知演讲者

08:45

让我来。

用户头像

奥菲尔弗里德

08:47

再次显示。

08:58

所以这是你的备忘单或指导方针，这纯粹是一个温和的指导方针或建议，请随意。

09:08

要完全不同地做任何事情，没有绝对正确的做事方式，有些会更容易，所以我会更难。

09:21

你应该使用你觉得最舒服的编程语言，你应该使用合作的风格，你觉得最舒服的方式。

09:30

所以。

09:32

我们应该怎么做。

09:34

好吧，让我首先告诉你我会设计什么方式，这不仅基于多年的教学，还基于来自开发的许多不同系统的许多不同的小组反馈，所以这里有一些关于如何去做的提示.

09:54

基本上，您的架构很简单，它有六个主要组件，它有一个查询输入管理器。

10:04

一种数据定义语言，一种以某种方式执行它的数据操作语言，在这种情况下它是一个主内存系统，而在这种情况下它是一个存储结构，它是一个磁盘，并且您需要形成某种程度的优化。

10:23

因此，我将详细描述其中的每一个，正如我所提到的，这不是必须这样做的，这只是一种更简单的方法，而且许多解决方案都是正确的。

10:38

所以查询输入管理器。

10:43

我看到了我多年来看到的人们花了很多时间而不是这个课程的目标是一个漂亮的查询输入管理器，他们有一个漂亮的界面，他们有一个很好的显示结果的方式，一切都很好.

11:01

但这不是这个类的目标，一个简单的行编辑器或简单的可视化编辑器就足够了，因为我们的目标是展示和理解关系数据库管理系统，它是如何操作的，而不是用户界面现在如何操作，如果你愿意的话就这样。

11:24

为了简化您的解析。

11:27

没关系，但是您的目标应该是确保您可以解析解决方案，您可以解析给出的查询，即按照您在项目二中所做的那样编写的 sql 查询。

11:45

你要做的就是，你必须找到sql语法。

11:49

记录语法是没有意义的，而且非常容易出错。

11:55

你可以在网上找到它，你可以以任何你想要的方式找到它，因此你可以像任何人一样依赖并使用任何你希望最简单的部分是递归下降的不良姿势。

12:11

你应该拿。

12:14

您应该使用递归的 disempower Sir 因为它更容易，但您不必这样做。

12:20

如果您找到了您希望使用的合作伙伴，您应该说明您使用的来源不一定要找到。

12:29

一个为你做这件事的系统，但是如果你看到我们解析或再次解析 sql 语言，这不是本课程的重点，我不是教你解析，因此这就是为什么说递归剥夺先生是绰绰有余。

12:49

If you do want a more sophisticated partial and you do find it online and is only doing your parsing not your execution.

12:58

feel free to take it state the source and make sure it's clearly demonstrated and stated in every DEMO that you do and in the right up that it is indeed taken code from some other place failure to do so is a violation.

13:16

site failure to do so if you take the code is a violation, you do not need to take their code, if you wish to code it yourself.

13:23

So that's a query input badger it doesn't it's not enough to be sophisticated it just should take his input, the sql statement and parse it into some kind of tokens.

13:33

Now.

13:35

The second unit should deal with data definition language, there are no actual data processed here, this is where you define your main entities that you're going to start If you recall.

13:50

The the database management system had.

13:54

A two tables, the main two tables was a tables table and a comms table, you will need to implement exactly those because at the end of the day, I should be able to say what tables are defined, and it should tell me, and what columns are in each table, and it should tell me as well.

14:14

And having done that will simplify your life significantly So the first thing you need to do is define and create a tables table and a comms table that you will use later on.

14:28

You must implement the following four entities, I do not require you to implement security, so you do not have to grant or revoke the security issues.

14:42

I do require you obviously to have relational tables, hence, you need to be able to create a table that's one instruction, you must be able to parse.

14:51

And hence you need to be able to enter the name of the table into the tables relation and every single attribute into the columns.

15:00

relation, including the identification of what is a foreign key what is a primary key and so on and so forth, the definition of what is know what is not know.

15:11

And that needs to be put in the create table when you put in the tables and calm relation.

15:18

If you create a table, you must be able to drop a table, so you need to be able to parse the drop table command.

15:26

I am requiring you to have indices indices means you need to be able to define what you index hands, you need to be able to parse the create index.

15:38

And hence you should be able to drop the index, you will I, you will notice, based on the data set that I will provide later on.

15:47

that it will be clear if you're using indices or you're not using indices, in order to find and do a search operation.

12:49

如果您确实想要一个更复杂的部分并且您确实在网上找到它并且只是在进行解析而不是执行。

12:58

随意将其说明来源，并确保在您所做的每个 DEMO 中清楚地证明和说明它确实是从其他地方获取的代码，如果不这样做是违规行为。

13:16

网站不这样做如果你拿代码是违规的，你不需要拿他们的代码，如果你想自己编码。

13:23

所以这是一个查询输入獾，它不够复杂，它只应该接受他的输入，sql语句并将其解析为某种标记。

13:33

现在。

13:35

第二个单元应该处理数据定义语言，这里没有处理实际数据，如果你记得的话，这是你定义你将要开始的主要实体的地方。

13:50

数据库管理系统有了。

13:54

两个表，主要的两个表是一个表表和一个通信表，您需要完全实现这些表，因为在一天结束时，我应该能够说出定义了哪些表，它应该告诉我，并且每个表中有哪些列，它也应该告诉我。

14:14

这样做会大大简化你的生活所以你需要做的第一件事是定义和创建一个表和一个你以后将使用的通信表。

14:28

您必须实现以下四个实体，我不要求您实现安全性，因此您不必授予或撤销安全性问题。

14:42

我确实要求你有关系表，因此，你需要能够创建一个表，它是一个指令，你必须能够解析。

14:51

因此，您需要能够将表​​的名称输入到表关系中，并将每个属性输入到列中。

15:00

关系，包括识别什么是外键什么是主键等等，定义什么是知道什么是不知道。

15:11

当您放入表格并保持关系时，需要将其放入创建表中。

15:18

如果创建表，则必须能够删除表，因此需要能够解析 drop table 命令。

15:26

我要求你有索引索引意味着你需要能够定义你的索引手，你需要能够解析创建索引。

15:38

因此，您应该能够根据我稍后提供的数据集删除索引。

15:47

如果您正在使用索引或者您没有使用索引，将会很清楚，以便查找和执行搜索操作。

15:55

For example, I may give you to find a couple on a large relation and that large relation will be found very quickly if you use an index and i'll take a long time, if you have to go sweep to the entire relation.

16:10

Now, in this stables and indices can be on many different things they can be on a single attribute, as you know, they can be on a variety of attributes, as you know.

16:21

I am not requiring to do multiple attribute indices, you will need to only do single attribute indices, and you should be able to do so on the key and some any other.

16:35

attribute that I asked you to do so, you should be able to do so on the fly I if I asked you to do so, I will probably have you do it i'm a small relation, because this could take time, but I may tell you to pre index, the large relation I had it.

16:53

Any index structure, you wish is sufficient for me again, this is not a course in data structures in a data structures course i'd have you do some kind of wide variety of structures, some kind of B plus tree.

17:07

Maybe red black tree, maybe two three trees some kind of a infamous form of a tree or try.

17:14

But in this case, I will take any type of index that you support and in any type of index will make it much quicker than without using an index.

17:22

So the data definition language that you must support our create table drop table create index and drop index.

17:32

Where the create table will put every relation table into a tables relation and the every attribute in the columns relation with the appropriate components in each of these tables and you must be able to create an index and dropping.

17:48

Those are the operations, you must support in the data definition language.

17:53

Data manipulation language actually involves real data and therefore you need to be able to insert into your table.

18:00

delete into your table and update your table, you must be able to deal with both single at single element insertion and bulk insertion if you do not have a way of bulk inserting.

18:15

Namely, in entire relation and one insertion and you actually call and load everything else it will take a long time for you to add material.

18:25

So do think about how to insert in the like when you do so, and delete when you do so, you must keep in mind the operation that we have discussed.

18:35

Remember, a relational database is based on sets and therefore sets have no duplicates so you need to be able to check for duplicates.

18:44

And when you delete you need to be able to maintain the referential integrity, we discussed in class.

18:50

So if I tell you delete on cascade you will delete on cascade if I tell you that you can delete and restrict, then you may not be able to delete if there's a situation where it calls for you not being able to delete.

19:02

keep in mind indexes complicate matters so when you delete an element, you have to make sure you maintain your index correctly when you are doing so.

19:13

So that's the data manipulation language again the three operators, I assume you're going to implement our insert delete an update.

19:25

We know the select operations pretty complex, we know that it's got a, however, a very rigid structure and that rigid structure makes it much easier to parse the components.

19:37

So whenever you get a select statement you need to be able to one parse the component to validate the table selection if I say select.

19:48

star from apple table, and there is no apple table that is a violation, you do not want your system to crash, that means you say no such table exists.

20:01

You need to be able to project on the attributes listed, including, if you want to list star if I tried to project, an attribute it doesn't exist, again, that is a problem.

20:13

Make sure you understand where the operators and what are the operand when do you have a select clause in a where there's a where condition, where a condition one and condition to and as an operator.

20:30

And the operand are the two things that you are quote unquote Andy.

20:35

identify the appropriate index says if you're using them and, as I said, you eventually we should I use some form of optimization.

20:46

optimization is the last thing you need to worry about right now, you need to be able to start this effort, and only later on, be able to optimize and you should have no problem, doing so once the infrastructure is done if it's designed well.

21:01

Once you've done all that you need to execute the query tree so that's the data manipulation language.

例如，我可以让你在一个大关系上找到一对夫妇，如果你使用索引，这个大关系会很快找到，如果你必须扫描整个关系，我会花很长时间。

16:10

现在，在这个马厩和索引可以在许多不同的事物上，它们可以在单个属性上，如您所知，它们可以在多种属性上，如您所知。

16:21

我不需要做多个属性索引，你只需要做单个属性索引，你应该能够在键和其他一些上这样做。

16:35

我要求你这样做的属性，你应该可以即时这样做我如果我要求你这样做，我可能会让你这样做我是一个小关系，因为这可能需要时间，但我可能会告诉你预先索引，我有它的大关系。

16:53

任何索引结构，你希望再次对我来说就足够了，这不是数据结构课程中的数据结构课程，我会让你做一些各种各样的结构，某种 B 加树。

17:07

也许是红黑树，也许是两三棵树，某种臭名昭著的树或尝试一下。

17:14

但是在这种情况下，我将采用您支持的任何类型的索引，并且在任何类型的索引中都会比不使用索引快得多。

17:22

所以你必须支持我们的create table drop table create index和drop index的数据定义语言。

17:32

创建表会将每个关系表放入表关系中，并且列关系中的每个属性都与这些表中的每个表中的适当组件相关，并且您必须能够创建索引和删除。

17:48

这些是操作，您必须在数据定义语言中支持。

17:53

数据操作语言实际上涉及真实数据，因此您需要能够插入到您的表中。

18:00

删除到您的表格并更新您的表格，如果您没有批量插入的方法，您必须能够处理单个元素插入和批量插入。

18:15

也就是说，在整个关系和一次插入中，您实际上调用并加载了其他所有内容，您需要很长时间才能添加材料。

18:25

所以在做的时候要考虑如何在like中插入，在做的时候删除，一定要记住我们讨论过的操作。

18:35

请记住，关系数据库基于集合，因此集合没有重复项，因此您需要能够检查重复项。

18:44

而当你删除时你需要能够保持参照完整性，我们在课堂上讨论过。

18:50

因此，如果我告诉您在级联上删除，如果我告诉您可以删除和限制，您将在级联上删除，那么如果出现需要您无法删除的情况，您可能无法删除。

19:02

请记住索引使事情变得复杂，因此当您删除一个元素时，您必须确保在这样做时正确维护您的索引。

19:13

这就是数据操作语言的三个操作符，我假设你要实现我们的插入删除更新。

19:25

我们知道选择操作非常复杂，但是我们知道它有一个非常严格的结构，并且这种严格的结构使得解析组件变得更加容易。

19:37

因此，每当您获得一条选择语句时，如果我说选择，您需要能够解析组件以验证表选择。

19:48

从苹果表中加星，并且没有苹果表是违规的，你不希望你的系统崩溃，这意味着你说不存在这样的表。

20:01

您需要能够在列出的属性上进行投影，包括，如果您想在我尝试投影时列出星号，则它不存在的属性再次出现，这是一个问题。

20:13

确保您了解操作符在哪里以及操作数是什么，您何时在哪里有一个选择子句，其中有一个 where 条件、一个条件和条件以及作为一个操作符。

20:30

操作数是你引用的两件事，不引用安迪。

20:35

确定适当的索引表示您是否正在使用它们，并且正如我所说，您最终我们应该使用某种形式的优化。

20:46

优化是您现在最不需要担心的事情，您需要能够开始这项工作，然后才能进行优化，并且您应该没有问题，一旦基础设施设计好就这样做.

21:01

一旦你完成了所有你需要执行的查询树，这就是数据操作语言。

21:11

Normally I would tell you a proper relational database management system would have the problem that deals with storage.

21:20

And amount of memory clearly as we just said and we discussed when we did cost bass evaluation, if you remember when I talked about inner loop and outer loop, we talked about memory blocks and how much memory, you can bring in and bring out of the inner and the outer relation.

21:39

i'm making your life a lot easier everything that you run assumes that it can be in main memory.

21:45

Therefore, you do not have to worry about blocking factors and you do not have to worry about memory exceptions, you can use as much memory, as you require, and therefore it will make your life a lot easier.

21:58

You should, however, know that you should be able to realize what is the inner relation and what is the outer relation in inner outer join and nested loop join.

22:08

And you should be able to know what about conjunct of in disjunctive clauses, because we all know that if you're looking for conjunct Dave you're trying to find the first false.

22:18

Because it's shortcuts and if you're looking for disjunctive you're trying to find the first true because it immediately we a results to true.

22:27

Obviously you don't want the big relation to be the outer relation.

22:33

As you recall we've we studied that.

22:39

that's a form of optimization however that's an easy form of optimization why because you will know exactly what is a large relation, which is an inner a, which is a small relation.

22:50

Because you're going to maintain that in your tables and you're going to know if it's an in conjunction were disjunctive operation because you're going to be doing the parsing.

23:02

You should, if you want to further process, you want to do some kind of efficient sorting immediately, as we discuss when somebody said, what does an amazing a efficient sword everybody jumps up and says oh that's got to be the quick source.

23:18

It is the quick sort if you don't know much about what's going on with your relation, however.

23:27

If you know the relation is mostly sorted because of the prior operation or because it's stored in some efficient way and you're using a key or whatever.

23:38

If it's mostly sorted and insertion sort is a basically nearly linear it is much easier and much faster.

23:46

And, of course, the type of join that we wish to do is either based on a sorted join or a nested loop joint depending on which is faster.

23:58

And that depends on our the relations of same size, in which case you want to use in a sword merge joint or are they very dramatically insides, in which case you want the nested loop, and as we remember the outer one being the smaller one.

24:15

So all this you're going to execute in main memory.

24:19

And it's good because you're going to be able to then store everything on a disk you should have the ability to store in a file and read from a file, there are different ways of showing reliability and efficiency and there's a question of delayed rights.

21:11

通常我会告诉你一个适当的关系数据库管理系统会遇到处理存储的问题。

21:20

和内存量很清楚，正如我们刚才所说的，我们在进行成本低音评估时讨论过，如果你记得当我谈到内循环和外循环时，我们谈到了内存块和多少内存，你可以带入和带出内在和外在的关系。

21:39

我让你的生活变得更轻松，你运行的所有东西都假设它可以在主内存中。

21:45

因此，您不必担心阻塞因素，也不必担心内存异常，您可以根据需要使用尽可能多的内存，因此它会让您的生活变得更轻松。

21:58

但是，您应该知道，您应该能够了解什么是内部关系，什么是内部外部联接和嵌套循环联接中的外部关系。

22:08

你应该能够知道在分离从句中的 conjunct of 是什么，因为我们都知道，如果你在寻找连词 Dave，你会试图找到第一个 false。

22:18

因为它是捷径，如果你正在寻找析取词，你试图找到第一个真，因为它立即我们的结果为真。

22:27

显然，您不希望大关系成为外部关系。

22:33

正如你所记得的，我们已经对此进行了研究。

22:39

这是一种优化形式，但这是一种简单的优化形式，为什么因为你会确切地知道什么是大关系，这是一个内部 a，它是一个小关系。

22:50

因为你要在你的表中维护它并且你会知道它是否是一个联合是析取操作，因为你要进行解析。

23:02

你应该，如果你想进一步处理，你想立即进行某种有效的排序，正如我们讨论的那样，当有人说，一把神奇的高效剑是什么，每个人都跳起来说哦，那一定是快速来源。

23:18

但是，如果您对您的关系中发生的事情知之甚少，这是快速排序。

23:27

如果您知道该关系主要是由于先前的操作或因为它以某种有效的方式存储并且您正在使用密钥或其他方式而对关系进行排序。

23:38

如果它主要是排序的并且插入排序基本上是线性的，那么它会更容易和更快。

23:46

当然，我们希望执行的连接类型是基于排序连接或嵌套循环连接，具体取决于哪个更快。

23:58

这取决于我们相同大小的关系，在这种情况下，您要在剑合并关节中使用，或者它们是否非常显着地在内部，在这种情况下，您需要嵌套循环，并且我们记得外部循环是较小的循环.

24:15

所以所有这些你都将在主内存中执行。

24:19

这很好，因为您将能够将所有内容存储在磁盘上，您应该能够存储在文件中并从文件中读取，有不同的方式来显示可靠性和效率，并且存在延迟权利的问题.

24:37

or packed blocks for various insertion efficiency or storage and efficiency.

24:43

i'm not requiring any of that you can, and you can explain to me what you did, and of course you'll get benefit to the doubt, but all these benefits of the doubt are nice only after your project is fully complete do not spend time trying to optimize before your project is running.

25:05

there's an old saying that you don't try to run the marathon before you launch even how to crawl currently you're not even crawling.

25:14

Once you're crawling you can walk once you're walking you can run and only then you worry about running the marathon so make sure that you develop it simply first before you go about doing so.

25:31

And the statement is implement via simplicity and then replace for a friend, she only wants it's working.

25:39

What are the potential optimizations you can do well, I had to do them, but let me make it more complete, we learned very early on.

25:49

In the optimization we learned that there are two types of optimizations that we typically push for.

25:55

One is called a rule based optimization and the other is called a cost based optimization.

26:02

We discussed that there are both types and we discussed, you can mix and match and, in fact, everything everybody does every system generally does.

26:10

The easiest of them for you to implement is called a rule based optimizing, it is the case that you push selections and projections down to the bottom, as quickly as you can.

26:23

there's one review involves maintaining almost no material no statistical information and therefore it is much easier to manage and handle.

26:32

It much tougher optimization to implement relies on maintaining statistics for whatever your relation you're storing trying to optimize.

26:41

And it basically deals with operation ordering based on activity and dynamic joint selection, namely inner versus outer space or a versus linear and so on and so forth operation along the way.

26:56

I sincerely doubt most teams every get to this point.

27:01

Every year, that I teach this or I signed this, there are some teams that do.

27:08

And there's always almost always I think there actually was always a team that so totally was me and what they've actually done.

27:19

And I hope that there is at least one such team that does so this time.

27:24

You will know that there is.

27:26

If there is one by the DEMO, you will see on the last day of class again, it will be optional if you're asked to DEMO.

27:35

So I would not worry about optimization just now I would worry about getting the simple things working first.

27:44

And remember there's a cardinal rule that's low, but functional, even if it's functionally poorly just completes the query processing definitely exceeds fast, but chaotic that does not work.

28:00

So with that.

28:04

Let me give you some words of warning.

28:08

There is no doubt that the first assignment was very relatively straightforward writing and collecting about making a presentation of data ethics did not take much time.

28:22

But it opened your eyes for what needs to be done, and we need to be liable and ethical about the data that we possess, and then we manipulate and handle.

28:33

Developing a database application, which is a very common assignment for an undergraduate class, in fact, a slightly more difficult assignment I gave you is generally the Semester project for an undergraduate database class.

28:48

They give a they make you do it user interface, which I did not do.

28:54

They make you runs to find some more tables which I had you define your own tables and they make you have a more common environment which I did not make you do, but all that is syntactic sugar on top of the hardcore database application, which I did make you do.

29:15

This is not such an assignment project three is a real assignment, it is an assignment that will significantly challenge the vast majority of you, and while i'm telling you this now, and most of you will ignore my warning.

24:37

或打包块用于各种插入效率或存储和效率。

24:43

我不需要你做的任何事情，你可以向我解释你做了什么，当然你会从怀疑中受益，但是只有在你的项目完全完成之后，怀疑的所有这些好处才是好的在项目运行之前不要花时间尝试优化。

25:05

有句老话，你不会在启动之前尝试跑马拉松，即使如何爬行目前你甚至不会爬行。

25:14

一旦你爬行，你就可以走路，一旦你走路，你就可以跑步，只有这样你才会担心跑马拉松，所以在你开始之前确保你先简单地开发它。

25:31

该语句通过简单实现，然后替换为朋友，她只希望它起作用。

25:39

你可以做好哪些潜在的优化，我不得不做，但让我让它更完整，我们很早就学会了。

25:49

在优化中，我们了解到我们通常会推动两种类型的优化。

25:55

一种称为基于规则的优化，另一种称为基于成本的优化。

26:02

我们讨论了这两种类型，我们讨论过，你可以混合搭配，事实上，每个人所做的一切，每个系统通常都会做。

26:10

其中最容易实现的称为基于规则的优化，即尽可能快地将选择和投影推到底部。

26:23

有一个审查涉及几乎不维护任何材料没有统计信息，因此它更容易管理和处理。

26:32

实现更困难的优化依赖于维护您存储的任何关系的统计信息以尝试优化。

26:41

它基本上处理基于活动和动态联合选择的操作排序，即内部与外部空间或a与线性等操作。

26:56

我真诚地怀疑大多数团队是否都能达到这一点。

27:01

每年，我教这个或者我签了这个，都有一些团队这样做。

27:08

而且几乎总是我认为实际上总是有一个团队完全是我和他们实际所做的事情。

27:19

我希望这次至少有一个这样的团队这样做。

27:24

你会知道有的。

27:26

如果DEMO有一个，你会在上课的最后一天再次看到，如果你被要求去DEMO，那将是可选的。

27:35

所以我现在不会担心优化，我会担心让简单的事情首先工作。

27:44

请记住，有一条基本规则很低，但很实用，即使它在功能上很差，只是完成查询处理肯定会超过快速，但混乱是行不通的。

28:00

所以就这样。

28:04

让我给你一些警告。

28:08

毫无疑问，第一个任务是相对简单的写作和收集关于数据伦理的介绍并没有花费太多时间。

28:22

但它让你看到了需要做什么，我们需要对我们拥有的数据承担责任和道德，然后我们操纵和处理。

28:33

开发一个数据库应用程序，这是一个本科班很常见的作业，其实我给你的一个稍微难一点的作业，一般是本科数据库班的Semester project。

28:48

他们提供了一个他们让你去做的用户界面，而我没有这样做。

28:54

它们让你跑去寻找更多的表，我让你定义自己的表，它们让你有一个更常见的环境，我没有让你这样做，但所有这些都是硬核数据库应用程序之上的语法糖，我确实让你做了。

29:15

这不是这样的任务 项目三是真正的任务，这是一项将对你们绝大多数人提出重大挑战的任务，虽然我现在告诉你们，但你们中的大多数人会忽略我的警告。

29:32

Let me go on record to tell you this isn't an assignment, you need to start now you have roughly a month to do so and based on past experience from other a groups, the amount of time that is usually takes to produce this assignment is somewhere around 50 to 75 hours of effort.

29:58

Considering that you have five weeks and consider your working groups that is reasonable.

30:06

However.

30:08

cramming it to the last week, even if you're three of you does not make things work there's a cardinal rule and software engineering that says that certain things can be sped up by more people certain things cannot.

30:26

and certain things can be done by cramming them together and certain things cannot a proper design, in this case will save you an enormous amount of effort later on.

30:38

I purposely do not define the specifics of this project, this is a project, you should be interacting with me, should you have any questions, it seems from past experience that those groups that asked for clarification early on, tend to do better.

30:56

Is it because they asked for clarification early on, or because they just simply started early on, I don't know and I never will.

31:05

But it's a word for the wise that's what you should do.

31:09

I want to thank you all for doing a solid job and projects one and projects to.

31:16

Not only did I find the presentations interesting I think many of the classmates learn new things, as did I based on the application develop, particularly for the covert application.

31:31

reminders reminder, this is not the same type of assignment, this is hard nuts and bolts implementation with proper design skills.

31:41

I wish you good luck with this assignment, I hope you started early, I urge you to start it early and i'm always happy to answer any questions you have on this assignment.

31:54

Good luck.

31:56

I will see you on class on Wednesday next week take care and again Good morning, good evening.

32:07

or good night.

29:32

让我记录下来告诉你这不是一个作业，你现在需要开始，你大约有一个月的时间来做这件事，根据其他小组过去的经验，完成这个作业通常需要的时间大约需要 50 到 75 小时的努力。

29:58

考虑到您有五周的时间并考虑您的工作组是合理的。

30:06

然而。

30:08

把它塞到最后一周，即使你们三个人也不能让事情顺利进行，有一条基本规则和软件工程说，某些事情可以被更多的人加速，某些事情不能。

30:26

有些事情可以通过塞在一起来完成，有些事情不能正确设计，在这种情况下，将为您节省以后的大量精力。

30:38

我故意不定义这个项目的细节，这是一个项目，你应该和我互动，如果你有任何问题，从过去的经验来看，那些早期要求澄清的小组往往做得更好。

30:56

是因为他们很早就要求澄清，还是因为他们只是很早就开始了，我不知道，我永远不会。

31:05

但这是明智的，这是你应该做的。

31:09

我要感谢大家做的扎实的工作和项目一和项目。

31:16

不仅我觉得演讲很有趣，我认为很多同学都学到了新东西，就像我基于应用程序开发的那样，特别是对于隐蔽应用程序。

31:31

提醒提醒，这不是同一类型的作业，这是具有适当设计技巧的硬性螺母和螺栓实施。

31:41

祝你这个作业好运，我希望你早点开始，我敦促你早点开始，我总是很乐意回答你关于这个作业的任何问题。

31:54

祝你好运。

31:56

下周星期三上课见，保重，早上好，晚上好。

32:07

或晚安。