



---

DataBase | HW01  
Fatemeh Nadi | 9636753

---

1.

A. B.

- **سجام** یا سامانه جامع اطلاعات مشتریان است.

که برای ارائه بهتر و سریع تر خدمات توسط شرکت سپرده گذاری مرکزی اوراق بهادار و تسویه وجوه آغاز به کار کرد. سجام یک سامانه زیر ساختی در بازار سرمایه است نه یک سامانه تجاری، زیرا هدف آن کسب درآمد نیست بلکه ایجاد یک سامانه مادر برای جمع آوری اطلاعات مشتریان است. این سامانه به صورت فراگیر برای ثبت نام اولیه تمامی سرمایه گذاران حقیقی، حقوقی، ایرانی و خارجی در بازار سرمایه استفاده می شود. بنابراین می توان آن را یک سامانه ملی نیز نام گذاری کرد.

در واقع این سامانه به احراز هویت افراد میپردازد و مرجع اصلی داده ها و اطلاعات سرمایه گذاران در بورس است در این سامانه مشتریان می توانند دارایی های خود مقدار سود یا زیان وارد شده به هر نماد را مشاهده کنند و همین طور به فروش اوراق خود بپردازند.

- افرادی که در سامانه سجام ثبت نام می کنند کفایت تنها یکبار فرم هایی که از سوی سازمان بورس تعریف شده را پر کنند. بنابراین در مراجعات بعدی خود به نهاد های مالی شامل صندوق های سرمایه گذاری، شرکت های تامین سرمایه و **کارگزاری ها**، دیگر نیازی به پر کردن همه فرم ها از اول نیست و روند بسیار کوتاه تری طی خواهد شد. زیرا تمام اطلاعات قبلا در سیستم ذخیره شده است. اطلاعات هویتی، شماره حساب، شماره موبایل، آدرس الکترونیکی و آدرس اقامت دائمی شما از طریق این سامانه ثبت می شود و از آن به بعد هیچ نهاد مالی نیازی به دریافت مجدد آن ندارد.

پس از ثبت نام **کد بورسی** خود را که کدی منحصر به فرد است و به صورت **primary key** در پایگاه داده داده قرار دارد برای یکتا کردن افراد را دریافت میکند.

اولین گام برای ورود به بورس و دریافت کد بورسی ثبت نام و احراز هویت در سامانه سجام است. در واقع مزیت اصلی ثبت نام و احراز هویت در سجام **sejam.ir** همین دریافت **کد معاملاتی** و ورود به بازار بورس است. کد بورسی یا معاملاتی مانند کد ملی برای هر فرد یکتا است و تنها یک بار برای هر فرد حقیقی صادر می شود .

جدولی برای نگهداری اطلاعات همه مشتری ها شامل اطلاعات تحصیلی و اطلاعات هویتی که پیش تر ذکر شد و همچنین محل اطلاعاتی از محل زندگی مشتری ها نیاز است که با کلید اصلی کد بورسی جدا می شوند و هیچ شخصی نمی تواند دو کلید بورسی داشته باشد این کلید شامل قسمتی از نام و نام خانوادگی مشتری به همراه عددی تصادفی هست

همچنین جدولی برای نگهداری اطلاعات کارگزاری ها که اشخاص با آن ها فعالیت بورسی خود را پیگیری میکنند نیاز است. این جدول شامل کد هر کارگزاری است ( کلید اصلی ) به همراه کد هر مشتری (کلید خارجی برای لینک به جدول قبلی ) که همان کد بورسی است و همچنین کلیدی که به ازاء هر کارگزاری برای مشتری ساخته میشود و به عنوان کد معاملاتی نگهداری می شود و به این ترتیب می توان با کد معاملاتی به کلیه کارگزاری هایی که یک کد بورسی (فرد) کار میکند پی برد

جدول دیگر مورد نیاز برای نگهداری های نماد های خریداری شده و یا فروخته شده فرد هست که شامل شاخص لحظه ای آن نماد (ارزش آن نماد) و مقدار سهام معامله شده و زمان دقیق انجام معامله باشد و همین شخص خریدار که برای این کار از کلید خارجی جدول میانی (کد معاملاتی) استفاده می شود حال چرا کد معاملاتی ؟ زیرا باید مشخص باشد در ساعت مقرر این سهام توسط چه کارگزاری برای مشتری (ها) خریداری شده است یا اینکه آیا خود شخص اقدام به خرید سهام و ایجاد معامله کرده است یا خیر همچنین باید مشخص شود که این معامله توسط چه کسی انجام شده در نتیجه کلیه اطلاعات شخص خریدار از قبیل اطلاعات اشخاص حقوقی که اقدام به معامله برای شخص دیگر میکنند نیز باید ثبت شود.

- کلیه اطلاعات سازمانها و داده های افراد در بورس تهران در پایگاه داده آن نگهداری میشود اطلاعاتی از قبیل داده های سازمان ها و شرکت هایی که سهام خود را به اشتراک گذاشته اند هست که نیاز به نگه داری شاخص لحظه ای و حجم عرضه شده نماد و کد آن نماد به عنوان کلید اصلی و کلید خارجی آن نماد هست سامانه سجام از اطلاعات این پایگاه داده برای محاسبه دارایی مشتری ها استفاده میکند همچنین سازمان بورس نیاز به پایگاه داده ای دارد برای نگه داری مقدار سهم هر مشتری که باید از کلید خارجی جدول اطلاعات مشتری در سامانه سجام که کد بورسی هست استفاده کند.

همچنین در کارگزاری های بورس به اطلاعاتی از قبیل اطلاعات افراد که شامل کد بورسی و کد معاملاتی آنها میشود دسترسی دارند که به سبب این دسترسی ها اقدام به خرید و یا فروش برای مشتری ها بکنند. در اینجا نیز نیاز به پایگاه داده ای داریم که شامل جدولی از اطلاعات هویتی افراد کلید اصلی که شامل دو فیلد کد بورسی و کد معاملاتی است و همچنین جدولی دیگر برای معاملات است اطلاعات معاملات انجام شده برای مشتری ها و سود دریافتی از هر معامله که با مشتری قرارداد بسته اند، نیاز دارند در این جدول باید اطلاعات نمادها خریداری شده شاخص لحظه ای و زمان دقیق خرید یا فروش آن اقدام کنند

در نتیجه برای ثبت این اطلاعات جداول این پایگاه داده باید با پایگاه داده سجام ارتباط داشته باشد تا تغییرات را ثبت نماید.

2.

A.

- Skill for Data Base Administrator(DBA)
  1. working with database software to find ways to store, organise and manage data
  2. DBAs must monitor the database performance and evaluate complex information coming from a variety of sources.
  3. Creating a backup and recovery strategy for the database, and regularly testing the backups to ensure usability.

- مهارت های تحلیلی – از مهمترین مهارت های مورد نیاز برای انجام درست کار در این شغل مهارت تحلیل است. با کمک این مهارت شما می توانید نیازهای مطرح شده توسط کارفرما یا مدیر پروژه را تحلیل کرده و ساختاری برای پایگاه داده طراحی کنید که نیازهای مذکور را پوشش دهد. همچنین از مهمترین ویژگی های مهم و ارزشمند پایگاه داده در سال های اخیر، رابطه ای بودن آن است. به این معنا که اجزای اطلاعاتی پایگاه داده شما به طرق مختلف با یکدیگر ارتباط دارند. به عنوان مثال در یک سیستم یکپارچه مالی و فروش در یک شرکت خصوصی لازم است اطلاعات مشتری به اطلاعات فاکتورهای فروش همان مشتری و اطلاعات فاکتورهای فروش به اطلاعات اسناد حسابداری و اطلاعات اسناد حسابداری به اطلاعات چک های وصولی و .... ارتباط داشته باشد. این ارتباط ها گاهی یک به چند، برخی از اوقات چند به یک و گاهی چند به چند است. کارشناس پایگاه داده –
- توجه به جزئیات – از آنجا که در طراحی و نگهداری پایگاه داده با جزئیات بسیار زیادی روبرو هستیم که هر کدام از آنها می تواند بر نتیجه کار تاثیر گذار باشد، لذا توجه به آنها از مهارت های لازم برای انجام این شغل است.
- فکر منطقی – از مهمترین مهارت های مورد نیاز برای انجام درست کار در این شغل مهارت ریاضی و منطقی است. به این معنا که اگر شما دارای این مهارت نباشید امکان طراحی و ساخت یک پایگاه داده قدرتمند را نخواهید داشت. لذا اگر در درس ریاضی نمرات خوبی کسب نکرده اید و یا علاقه زیادی به آن ندارید به دنبال شغل دیگری باشید.
- مهارت های حل مساله – در این شغل همانند سایر مشاغل تخصصی، مهارت حل مساله موجب جلوگیری از سردرگمی و خلق راهکار های جدید در موقع مواجه شدن با مشکلات پیش آمده در کار می شود.

- Skill for Data Scientist(DS):

1. **Education**
2. **Teamwork**
3. **R Programming**
4. **Unstructured data**
5. **Python Coding**
6. **Hadoop Platform**
7. **SQL Database/Coding**
8. **Apache Spark**
9. **Machine Learning and AI**
10. **Data Visualization**
11. **Communication skills**

B.

- **Data Base Administrator :**

- 

in the United States is **\$93,316** as of September 25, 2020, but the salary range typically falls between **\$81,416** and **\$105,908**.

- آمریکا - متوسط درآمد سالیانه مدیران پایگاه داده مطابق با آخرین آمارها، \$67,733 است
- استرالیا - میزان درآمد سالانه مدیران پایگاه داده A\$86,357 / yr استرالیا (قبل از کسر مالیات) بوده است.
- انگلستان - متوسط درآمد سالیانه این شغل برای افراد تازه کار بین £48,507 - £436 است

- **Data Scientist :**

- Salary Range in micro Soft: INR 1,300,000 to INR 2,100,000
- The average salary for a Data Scientist is A\$16,000 and Range - A\$19 - A\$32,458
- The average salary for a Data Scientist is £50,461 in London, and range £606 - £10,937
- Italy \$60K
- France \$55K
- Japone \$75K

در حقیقت مدل تجسمی است از دنیای واقعی و اشیا و موجودیت هاست.  
 مدل کردن داده روشی برای توصیف داده ها و عملیات روی آنها در سطوح مختلف معماری پایگاه داده است.  
 اطلاعات هویتی افراد از قبیل سن و شغل

مدل کردن داده (data modeling) تحلیل و توصیف داده های محیط عملیاتی و ارتباط بین آنها و شرح معنی و قیدهای داده است.

یک مدل داده قالب قراردادی برای ساخت و کارکردن با داده دراختیار می گذارد.

یک مدل داده پدیده های دنیای واقعی را توصیف می کند و ساختار داده مورد نیاز برای حل مسئله خاص و ارتباط درونی بین ساختارها را تعریف می کند. دیاگرام ER مثالی از این نوع است که بیشتر توسط طراحان و تحلیل گران سیستم به کار می رود.

علاوه براین مدل داده تعیین می کند یک DBMS چگونه داده را درون خود، به کاربران و برنامه های کاربردی نمایش دهد. مدل رابطه ای مثالی برای این نوع از مدل داده است.

موارد زیر در مدلسازی بهتر به ما کمک خواهد کرد:

- سوابق بیماری شخص بیمار
- ذخیره آزمایشات 2 سال گذشته و برخی آزمایشات مهم
- تاریخ واکسیناسیون
- حساسیت دارویی
- پزشک(های) معالج
- اطلاعاتی درمورد پزشک
- وضعیت عمومی فرد
- تاریخ های مراجعه به پزشک

4.

disadvantages.

- *Increased costs.*
- *Management complexity*
- *Maintaining currency*
- *Vendor dependence*
- *Frequent upgrade/replacement cycles*

- با ایجاد مکانیزم پایگاه داده کار پیچیده زمانبر خواهد بود
- به امکانات سخت افزاری و نرم افزاری زیادتری نیازمند خواهیم بود
- با ایجاد لطمه ای به پایگاه داده کلیه اطلاعات از دست میبرود و اپلیکیشن های مربوطه مختل خواهد شد.

- برای تبدیل فایل به پایگاه داده هزینه و انرژی زیادی باید صرف کرد
- پیچیده شده سیستم افراد متخصص را برای کار با آنها می‌خواهد که این نیز برای سامان هزینه بر است.

5.

1. پشتیبانی از داده‌ها در انتیتی‌ها مثل اپلیکیشن
2. تست اپلیکیشن و داده‌ها برای موارد امنیتی داده و صحت اطلاعات حریم شخصی و یکپارچگی کل داده‌ها
3. نظارت و پشتیبانی و backup گیری
4. بازیابی اطلاعات
5. آنالیز و استفاده از داده‌ها در جهت نیازمندی‌ها و توزیع درست آنها
6. سازماندهی با توجه به استاندارد‌های موجود

6.

- A.      7      رکورد  
            5      فیلد

B.

چون برای شهر فیلد جداگانه‌ای در نظر گرفته نشده است در طراحی دیتا بیس و داخل فیلد manager\_address آورده شده نمی‌توان به راحتی روی شهر فیلتر گذاشت برای رفع این مشکل فیلد manager\_address را به فیلد های area code , state , city , street , zip\_code تبدیل می‌کنیم. و برای مثال سطر اول قسمت manager\_address را اینگونه ذخیره می‌کنیم :

Zip_code	Street	city	state	Area_code
3334	Lee Rd.	Galnesville	FL	37123

C.

یلد manager\_address را به فیلد های area code , state , city , street , zip\_code تبدیل می‌کنیم. و برای مثال سطر اول قسمت manager\_address را اینگونه ذخیره می‌کنیم :

Zip_code	Street	city	state
3334	Lee Rd.	Galnesville	FL

و area code ,manager phone باهم ادغام شوند که در صورت تغییر یکی دیگری نیز تغییر کند یعنی با دادن کد منطقه قسمت اول شماره تلفن نیز ست شود

D.

area code برای منطقه Galnesville اگر مقدار 904 باشد در رکورد اخر نیز Stetson area code نیز 904 خورده شده در صورتی که باید متفاوت باشد زیرا هر شهر Area code یکتا دارد.

ناهنجاری اینجا اینگونه به وجود می آید که یکبار باید نام شهر را وارد نمود و یکبار کد منطقه در صورتی که با ایجاد مکانیزی این دو را میتوان با یک دریافت اطلاعات دریافت نمود و ممکن است آدرس فرد تغییر کند اما شماره فرد تغییر نکند و خود باعث میشود area code تغییر نکند. و اینگونه مشکل ناهنجاری را رفع نمود با تنظر بین کد منطقه و نام شهر

7

A.

در سطر 8 به نام پروژه متفاوت از مابقی رکورد ها نوشته شده و یک | کمتر از بقیه دارد این مشکل در ادامه برای فیلتر گذاری روی این پروژه ما را به اشتباه می اندازد

B.

EMP\_NAME

باید به دو فیلد emp\_first-name , emp\_last\_name تبدیل کرد

EMP\_PHONE

باید به سه فیلد emp\_state\_phone , emp\_city\_phone , emp\_num\_phone تبدیل نمود.

اگر بتوان ساختار این جدول را به 3 جدول دیگر تبدیل کرد و با ارتباط بین جداول به هدف نهایی رسید که طراحی بهتری داشت

جدول یک برای کارمندان ( شماره کارمند , نام , نام خانوادگی , کد شغل )

جدول دوم جدول شغل ها

جدول سوم جدول پروژه

8

σyear=2018(section)

→ no result EMPITY

[[course\_id(σyear=2018(section))]]

→ no result EMPITY

[[course\_id(section)]

→

Course_id
105
137
158
169
192
200

$\rho_s(\Pi_{\text{course\_id}}(\text{section}) - \Pi_{\text{course\_id}}(\sigma_{\text{year}=2018}(\text{section})))$

→ Table name = s

Course_id
105
137
158
169
192
200

$\rho_c(\text{course})$

→ table name = c

	course_id [PK] character varying (8)	title character varying (50)	dept_name character varying (20)	credits numeric (2)
1	101	Diffusion and Phase Transfo...	Mech. Eng.	3
2	105	Image Processing	Astronomy	3
3	123	Differential Equations	Mech. Eng.	3
4	127	Thermodynamics	Geology	3
5	130	Differential Geometry	Physics	3
6	133	Antidisestablishmentarianis...	Biology	4
7	137	Manufacturing	Finance	3
8	139	Number Theory	English	4
9	158	Elastic Structures	Cybernetics	3
10	169	Marine Mammals	Elec. Eng.	3

$\rho_s(\Pi_{\text{course\_id}}(\text{section}) - \Pi_{\text{course\_id}}(\sigma_{\text{year}=2018}(\text{section}))) \bowtie_{s.\text{course\_id}=c.\text{course\_id}} \rho_c(\text{course})$

→

	Course_id	title	Dept_name	credits
1	105	Image processing	Astronomy	3
2	137	Manufacturing	Finance	3
3	158	Elastic Structures	Cybernetics	3
4	169	Marine Mammals	Elec Eng	3

$\Pi_{c.\text{course\_id}, c.\text{title}}(\rho_s(\Pi_{\text{course\_id}}(\text{section}) - \Pi_{\text{course\_id}}(\sigma_{\text{year}=2018}(\text{section}))) \bowtie_{s.\text{course\_id}=c.\text{course\_id}} \rho_c(\text{course}))$

→



	Course_id	title
1	105	Image processing
2	137	Manufacturing
3	158	Elastic Structures
4	169	Marine Mammals

9

به مثال زیر توجه کنید:

101 mon 9:30

101 sat 9:30

در مثال بالا از 101 به عنوان time\_slot\_id مورد استفاده قرار میگیرد که برای اتصال دو بازه زمانی به هم به کار می رود مثلاً یک کلاس 3 واحدی باید در طی ساعت در هفته ارائه شود که به دو بخش 1:30 تقسیم میکنیم حال برای لینک کردن این دو یا چند بخش نیاز به idهای یکسان داریم که تداخل زمانی پیش نیاید اما این time\_slot\_id به تنهایی نمیتواند به عنوان کلید اصلی قرار بگیرد زیرا همانطور که قبل تر ذکر شد دو فیلد دارای time\_slot\_id یکسانی هستند در جدول پس نیاز به پارامترهای بیشتری داریم با اضافه کردن فیلد روز می توان فیلتر کرد تعدادی از رکورد ها را از یک دیگر ولی باز هم امکان تعدد فیلد ها پیش می آید بدین صورت که یک کلاس یا time\_slot\_id 100 و روز mon فرض کنید که در ساعت 12:30 شروع به کار و بخش دیگر آن در روز sat و در همان ساعت 12:30 شروع به کار کند اگر این درس 4 واحدی باشد و بخش دیگر ارائه در ساعت 9 صبح روز mon باشد باز ما به دو فیلد در جدول میرسیم که day , time\_slot\_id یکسانی دارند با اضافه کردن فیلد Start\_time میتوان به کلید اصلی رسید.

100 mon 12:30

100 sat 9

100 sat 12:30

10

باید توجه داشت که نباید مقدار null را با مقدار صفر اشتباه گرفت اهمیت null را با این مثال پیگیری میکنم: فرض کنید پایگاه داده نمرات دانشجویان را در اختیار داریم اگر فیلد نمره شخص با Null پر شده باشد به معنای این است که هنوز نمره اعلام نشده یا در سامانه وارد نشده است اما اگر صفر وارد شده باشد یعنی شخص نمره 0 را کسب کرده است

از دیگر موارد کاربرد می توان برای محاسبه معدل فعلی گفت اگر به جای NULL از مقدار صفر استفاده کنیم معدل غیر واقعی نمایش داده میشود اما اگر Null را اعمال کنیم با شرط گذاشتن برای محاسبه معدل که اگر null بود جز نمرات محاسبه نگردد میتوان به معدل واقعی دست یافت یا در دیگر طراحی ها فضایی را متصور بشویم که تعدادی از فیلد ها هنوز پر نشده اند با شرط گذاشتن روی فیلد ها با مقادیر Null میتوان به این فیلد ها را پر کرد.

11

خیر زیرا در جدول instructor برای فیلد name دامینی محدود نشده است که نمیتوان دو اسمی را پیدا کرد که یکسان نباشند  
و این عدم محدودیت در دامنه name سبب میشود که بتوان رکوردی را درج نمود که نام تکراری داشته باشد در نتیجه ماهیت کلید اصلی زیر سوال میرود

12

- A. PK for branch : branch\_name  
PK for customer : customer\_name  
PK for loan : loan\_number  
PK for account : account\_number
- B. FK for loan : branch\_name  
FK for borrower : customer\_name and account\_name  
FK for account : branch\_name  
FK for depositor : customer\_name and loan\_number
- C.  
 $\Pi_{branch\_name}(\sigma_{branch\_city=chicago}(branch))$
- D.  
 $\Pi_{loan\_number}(\sigma_{amount>10000}(loan))$

13

A. Entity Supertypes

سوپرتایپ یک نوع موجودیت عمومی است که با یک یا چند زیرگروه رابطه دارد. یک موجودیت که با چند و شامل مشخصات مشترک از subtype های خود میشود

B. Entity Subtypes

Subtypes به زیر گروهی از Entity ها در یک نوع موجودیت گفته می شود که برای سازمان معنی دار است و ویژگی ها یا روابط مشترک متمایز از سایر زیر گروه ها را دارد. و شامل مشخصات یکتایی میباشد که با دیگر subtype ها متفاوت است.

#### C. Subtype Discriminator

یک subtype discriminator نوع ویژگی در موجودیت Supertypes است که تعیین می کند وقوع نوع فوق به کدام Subtype مربوط می شود.

#### D. Entity Clustering

یک نوع موجودیت "مجازی" است که برای نمایش چندین موجودیت و روابط در ERD استفاده می شود. یک خوشه نهادی با ترکیب چندین موجودیت بهم پیوسته در یک شی موجود انتزاعی منفرد شکل می گیرد.

14

- **یکتا بودن** : در تعریف PK آمده است که باید بتوان به وسیله کلید اصلی به تنها یک فیلد دست یافت
- **تغییر ناپذیر در طول زمان** : PK باید دائمی و غیرقابل تغییر باشد. باید ویژگی یا ویژگی هایی از یک ماهیت را انتخاب کرد که تغییر ناپذیر باشند در طول زمان و اگر آن ویژگی تغییر کرد به معنای تغییر ماهیت آن موجودیت باشد زیرا در صورت تغییر کلید اصلی باید تمام کلید های خارجی جداول دیگر نیز تغییر کند که کار را بسیار سخت میکند.
- **Security-compliant** : نباید کلید اصلی از الگوی خاصی تبعیت بکند زیرا حفره امنیتی ایجاد میکند و سبب میشود افراد به داده های غیر مجاز دسترسی پیدا کنند.
- **Preferably single-attribute** : لزومی وجود ندارد اما بسیار کار را راحتتر میکند اگر کلید غیرقابل کاهش باشد برای ارتباط بین جداول حجم محاسبات و کار پایگاه داده را کاهش میدهد.
- **Preferably numeric** : مقادیر منحصر به فرد را می توان هنگام عددی بودن بهتر مدیریت کرد زیرا روال مشخصی برای فیلد های جدید وجود خواهد داشت و اکثر سیستم ها این عددی بودن را هندل میکنند
- **Non-intelligent** : نمود بیرونی هوشمند نداشته باشد زیرا خود مدلی از یک موجودیت غیر هوشمند است.

15

composite primary key که در آن بیش از یک ستون برای تعیین کلید اصلی یک جدول استفاده می شود. در چنین مواردی ، کلیه کلیدهای خارجی نیز باید شامل تمام ستون ها در کلید ترکیبی باشند. و ستون هایی که یک کلید ترکیبی را تشکیل می دهند می توانند از انواع مختلف داده باشند. و در مواقع زیر کاربرد دارد.

- As identifiers of composite entities, in which each primary key combination is allowed only once in the M:N relationship
- As identifiers of weak entities, in which the weak entity has a strong identifying relationship with the parent entity

16

کلید اصلی را surrogate key می گوئیم که مقدار آن به صورت اتوماتیک زیاد شود در حقیقت این کلید معادل حقیقی در دنیای واقعی ندارد.

surrogate key کلید اصلی است که توسط طراح پایگاه داده ایجاد می شود و برای تمییز دادن یک موجودیت ( فیلد ) از دیگری است. ( جستجوی بهتر یا ارتباط جداول نیز میتواند کاربرد داشته باشد). برای تعریف دقیق تر به مقایسه کلید اصلی با surrogate key می پردازم:

- Both keys contain a unique value for a record in a table.
- Primary keys hold some business meaning whereas surrogate does not hold any business meaning.
- Primary key may contain numeric as well as non-numeric values whereas surrogate keys contain only (simple) numeric values.
  - Surrogate keys are simple numeric values, as simple as normal counting. So most of the time they save storage space.
  - As surrogate keys are simple and short, it speed-up the join performance.

کاربرد این نوع از کلید برای تمییز دادن دو موجودیت از یک دیگر است و چون وابستگی به داده ندارد اگر جایی بنابر نیاز بخواهیم پایگاه داده را تغییر دهیم استفاده از این کلید کار ما را راحتتر میکند.

همچنین کوتاهی کلید سبب میشود که کد کمتری برای رسیدن به انتیتی خاص اجرا شود.

از دیگر کاربرد ها هنگام جابجایی داده ها بین سیستم های آزمایشی ، دیگر نگران کلیدهای تکراری نیستیم زیرا با درج داده ، کلید جدید تولید می شود.