

# תרגילים למתחילים - SQLAlchemy עם MySQL

## הוראות כלליות

- כל תרגיל בנוי צעד אחר צעד
- נסה לפתור לפי הסדר
- השתמש במדריך אם נתקעת
- בדוק את הקוד עם נתוני בדיקה

## חלק 1: התחלה - מודלים ראשוניים

### תרגיל 1.1: מודל ספר פשוט

צור מודל Book עם SQLAlchemy.

שדות נדרשים:

- id - מפתח ראשי, אוטומטי
- title - כותרת הספר (חובה, מקסימום 200 תווים)
- author - שם המחבר (חובה, מקסימום 100 תווים)
- pages - מספר עמודים (מספר שלם)
- price - מחיר (מספר עשרוני)

דרישות:

- הגדר `tablename__ = "books__"`
- השתמש ב-Field עם הגבלות מתאימות

### תרגיל 1.2: חיבור ל-MySQL

צור חיבור לבסיס נתונים MySQL.

דרישות:

- צור בסיס נתונים בשם `bookstore_db`
- התחבר עם `mysql+pymysql`
- צור את הטבלה `books`
- הדפס הודעת הצלחה

רמז: השתמש ב-`create_engine` ו-`create_all`

### תרגיל 1.3: הוספת ספר ראשון

כתוב פונקציה `add_book(title, author, pages, price)` שמוסיפה ספר חדש.

דרישות:

- השתמש ב-Session
- שמור את הספר עם `commit`

- הדפס את ה-id של הספר שנוצר
- החזר את אובייקט הספר

## תרגיל 1.4: הצגת כל הספרים

כתוב פונקציה `show_all_books()` שמציגה את כל הספרים.

דרישות:

- השתמש ב-`select(Book)`
- הדפס כל ספר בפורמט: ID: X | כותרת: Y | מחבר: Z | עמודים: W | מחיר: V
- אם אין ספרים, הדפס הודעה מתאימה

## תרגיל 1.5: חיפוש ספר לפי ID

כתוב פונקציה `get_book_by_id(book_id)` שמחזירה ספר ספציפי.

דרישות:

- השתמש ב-`session.get()`
- אם הספר לא נמצא, הדפס הודעה והחזר `None`
- אם נמצא, הדפס את כל הפרטים

## תרגיל 1.6: עדכון מחיר ספר

כתוב פונקציה `update_book_price(book_id, new_price)` שמעדכנת מחיר.

דרישות:

- מצא את הספר לפי ID
- עדכן את המחיר
- שמור את השינויים
- הדפס הודעה על הצלחה או כישלון

## תרגיל 1.7: מחיקת ספר

כתוב פונקציה `delete_book(book_id)` שמוחקת ספר.

דרישות:

- מצא את הספר
- מחק אותו עם `session.delete()`
- הדפס הודעה מתאימה

## תרגיל 1.8: ספירת ספרים

כתוב פונקציה `count_books()` שמחזירה כמה ספרים יש במערכת.  
רמז: השתמש ב-`len()` על רשימת הספרים

## תרגיל 1.9: הוספת ספרים מרשימה

כתוב פונקציה שמקבלת רשימה של ספרים ומוסיפה אותם.

דוגמה לקלט:

`books =`

("הארי פוטר", "ג'יי קיי רולינג", 400, 79.90),

("שר הטבעות", "טולקין", 600, 89.90),  
("1984", "ג'ורג' אורוול", 350, 59.90)

[

דרישות:

- עבור על הרשימה ב-loop
- הוסף כל ספר
- הדפס כמה ספרים נוספו

## תרגיל 1.10: בדיקת קיום ספר

כתוב פונקציה `book_exists(title)` שבודקת אם ספר עם כותרת מסוימת קיים.

דרישות:

- השתמש ב-`select().where()`
- החזר `True` או `False`

## חלק 2: סינון וחיפוש

### תרגיל 2.1: חיפוש ספרים לפי מחבר

כתוב פונקציה `find_books_by_author(author_name)` שמחזירה את כל הספרים של מחבר.

דרישות:

- השתמש ב-`where(Book.author == author_name)`
- הצג את התוצאות בצורה מסודרת

### תרגיל 2.2: ספרים מתחת למחיר

כתוב פונקציה `get_cheap_books(max_price)` שמחזירה ספרים מתחת למחיר מסוים.

דרישות:

- השתמש ב-`where(Book.price < max_price)`
- מיין לפי מחיר (מהזול ליקר)

### תרגיל 2.3: ספרים ארוכים

כתוב פונקציה `get_long_books(min_pages)` שמחזירה ספרים עם יותר מ-X עמודים.

דרישות:

- סינון לפי מספר עמודים
- מיין לפי מספר עמודים (מהרבה לפחות)

### תרגיל 2.4: חיפוש חלקי בכותרת

כתוב פונקציה `search_books(keyword)` שמחפשת ספרים שהכותרת מכילה מילת מפתח.  
רמז: השתמש ב-`Book.title.contains(keyword)`

### תרגיל 2.5: טווח מחירים

כתוב פונקציה `books_in_price_range(min_price, max_price)` שמחזירה ספרים בטווח מחירים.

דרישות:

- `price >= min_price AND price <= max_price`

### תרגיל 2.6: הספר היקר ביותר

כתוב פונקציה `get_most_expensive_book()` שמחזירה את הספר היקר ביותר.  
רמז: השתמש ב-`order_by(Book.price.desc()).first()`

### תרגיל 2.7: הספר הזול ביותר

כתוב פונקציה `get_cheapest_book()` שמחזירה את הספר הזול ביותר.

### תרגיל 2.8: חישוב ממוצע מחיר

כתוב פונקציה `calculate_average_price()` שמחשבת את המחיר הממוצע של כל הספרים.

רמז: קרא את כל הספרים וחשב ממוצע בעצמך

## תרגיל 2.9: ספרים לפי אורך

כתוב פונקציה `get_books_sorted_by_length(ascending=True)` שמחזירה ספרים ממוינים לפי מספר עמודים.

פרמטר:

- `ascending=True` - מהקצר לארוך
- `ascending=False` - מהארוך לקצר

## תרגיל 2.10: דפדוף (Pagination)

כתוב פונקציה `get_books_page(page_number, page_size=10)` שמחזירה דף מסוים של ספרים.

דוגמה:

- דף 1, גודל 10: ספרים 1-10
- דף 2, גודל 10: ספרים 11-20

רמז: השתמש ב-`offset()` ו-`limit()`

## חלק 3: אימות נתונים

### תרגיל 3.1: שיפור מודל Book עם אימותים

שפר את מודל Book עם אימותים:

דרישות:

- title - לפחות 2 תווים
- author - לפחות 2 תווים
- pages - בין 1 ל-5000
- price - בין 0.01 ל-999.99

רמז: השתמש ב-Field(ge=, le=, min\_length=)

### תרגיל 3.2: שדות אופציונליים

הוסף שדות נוספים ל-Book:

- isbn - אופציונלי, מקסימום 20 תווים, ייחודי
- publication\_year - אופציונלי, בין 1000 ל-2030
- in\_stock - בוליאני, ברירת מחדל True

### תרגיל 3.3: תאריך הוספה אוטומטי

הוסף שדה created\_at שמתמלא אוטומטית בתאריך ושעה נוכחיים.

רמז: השתמש ב-Field(default\_factory=datetime.now)

### תרגיל 3.4: בדיקת ISBN ייחודי

כתוב פונקציה add\_book\_with\_isbn(title, author, pages, price, isbn) שבודקת שה-ISBN לא קיים לפני הוספה.

### תרגיל 3.5: עדכון מלאי

כתוב פונקציות:

- mark\_out\_of\_stock(book\_id) - מעדכן in\_stock=False
- mark\_in\_stock(book\_id) - מעדכן in\_stock=True

### תרגיל 3.6-3.10

3.6: רשימת ספרים זמינים - כתוב פונקציה get\_available\_books()

3.7: עדכון מרובה - פונקציה לעדכון כמה שדות בבת אחת

3.8: סטטיסטיקות ספרים - פונקציה עם מידע מסכם

3.9: ספרים שפורסמו בעשור - סינון לפי טווח שנים

3.10: ניקוי מלאי ישן - מחיקת ספרים ישנים

## חלק 4: קשרים בין טבלאות

### תרגיל 4.1: מודל Category

צור מודל Category עם השדות:

- id
- name - שם הקטגוריה (ייחודי, מקסימום 50 תווים)
- description - תיאור (אופציונלי)

### תרגיל 4.2: קשר Category ← Book

עדכן את Book להכיל קשר ל-Category:

דרישות:

- הוסף category\_id (מפתח זר)
- הוסף category (Relationship)
- עדכן את Category עם רשימת books

### תרגיל 4.3: יצירת קטגוריות

כתוב פונקציה add\_category(name, description) שמוסיפה קטגוריה.  
צור לפחות 5 קטגוריות:

- מדע בדיוני
- פנטזיה
- מסתורין
- רומנטיקה
- ביוגרפיה

### תרגילים 4.4-4.10

4.4: הוספת ספר עם קטגוריה

4.5: ספרים לפי קטגוריה

4.6: הצגת קטגוריה עם ספרים

4.7: העברת ספר לקטגוריה אחרת

4.8: קטגוריות פופולריות

4.9: מחיקת קטגוריה ריקה

4.10: מודל Author נפרד

## חלק 5: מערכת השאלות ספרים

### תרגיל 5.1: מודל Customer

צור מודל Customer עם:

- id
- name - שם מלא
- email - ייחודי
- phone - אופציונלי
- registration\_date - תאריך הרשמה אוטומטי

### תרגיל 5.2: מודל Loan

צור מודל Loan עם שדות למעקב אחר השאלות ספרים, כולל תאריכי השאלה והחזרה.

### תרגילים 5.3-5.10

- 5.3: הוספת לקוח
- 5.4: השאלת ספר
- 5.5: החזרת ספר
- 5.6: היסטוריית השאלות של לקוח
- 5.7: ספרים מושאלים כרגע
- 5.8: ספרים באיחור
- 5.9: לקוחות פעילים
- 5.10: דוח חודשי



## חלק 6: מערכת Reviews (ביקורות)

### תרגילים 6.1-6.10

- 6.1: מודל Review - צור מודל לביקורות עם דירוג ותגובה
- 6.2: הוספת ביקורת - פונקציה להוספת ביקורת חדשה
- 6.3: ביקורות של ספר - הצגת כל הביקורות לספר
- 6.4: דירוג ממוצע - חישוב דירוג ממוצע לספר
- 6.5: ספרים מדורגים גבוה - סינון ספרים לפי דירוג
- 6.6: הוספת דירוג למודל Book - property לחישוב דירוג
- 6.7: ביקורות של לקוח - היסטוריית ביקורות
- 6.8: עדכון ביקורת - שינוי דירוג או תגובה
- 6.9: מחיקת ביקורת
- 6.10: דוח ביקורות - סטטיסטיקות כלליות

## חלק 7: פונקציות מתקדמות

### תרגילים 7.1-7.10

- 7.1: ייצוא לקובץ - ייצוא ספרים לקובץ טקסט
- 7.2: ייבוא מרשימה - ייבוא ספרים מקובץ
- 7.3: חיפוש מתקדם - חיפוש לפי מספר פרמטרים
- 7.4: גיבוי בסיס נתונים - ייצוא ל-JSON
- 7.5: שחזור מגיבוי - ייבוא מ-JSON
- 7.6: מערכת המלצות פשוטה - המלצות מבוססות קטגוריות
- 7.7: טבלת סטטיסטיקות - הצגת נתונים מסודרת
- 7.8: ניקוי אוטומטי - מחיקת נתונים ישנים
- 7.9: לוח בקרה CLI - תפריט אינטראקטיבי
- 7.10: פרויקט מלא - מערכת חנות ספרים מלאה

## טיפים להצלחה

- ✓ **תכנן את המבנה** - שרטט את הטבלאות והקשרים על נייר תחילה
- ✓ **בדוק שלב אחר שלב** - הרץ כל פונקציה מיד אחרי שאתה כותב
- ✓ **השתמש בנתוני דוגמה** - צור כמה ספרים ולקוחות לבדיקה
- ✓ **טפל בשגיאות** - תמיד בדוק אם דברים קיימים לפני פעולות
- ✓ **כתוב קוד נקי** - שמות משתנים ברורים, הערות, docstrings
- ✓ **שמור גרסאות** - commit לאחר כל תרגיל שעובד
- ✓ **בקש עזרה** - תיעוד, פורומים, מדריכים

## בדיקה עצמית

לאחר שסיימת את התרגילים, ודא ש:

- ✓ אתה יכול ליצור מודלים עם SQLAlchemy
- ✓ אתה מבין קשרים (Relationships)
- ✓ אתה יכול לבצע CRUD (Create, Read, Update, Delete)
- ✓ אתה יודע לסנן ולחפש נתונים
- ✓ אתה יכול לעבוד עם MySQL
- ✓ אתה מטפל בשגיאות בצורה נכונה
- ✓ הקוד שלך נקי וקריא

## משאבים מומלצים

תיעוד: 

- SQLModel Documentation
- SQLAlchemy Documentation
- Pydantic Documentation

וידאו: 

- SQLModel Tutorial by FastAPI creator
- MySQL with Python tutorials

קהילות: 

- sqlmodel תגית - Stack Overflow
- Reddit r/learnpython
- Discord servers for Python

 **בהצלחה!**

זכור: העיקר זה לתרגל. אל תתייאש אם משהו לא עובד - זה חלק מהתהליך!