

פרויקט: מנתח לוגים של תעבורת רשת

המשימה

יחידת המודיעין קיבלה קובץ לוג ענק - אלפי רשומות של תעבורת רשת מהשבועות האחרונים. יש חשד שגורם עוין מנסה לחדור למערכת. המשימה שלכם: לבנות מערכת אוטומטית שתסרוק את הנתונים, תזהה דפוסים חשודים, ותייצר דוח מסודר עם כל האיומים הפוטנציאליים. הזמן לוחץ - אי אפשר לעבור ידנית על 10,000 שורות. אתם צריכים קוד חכם שיעשה את העבודה.

רקע: מה זה פורט?

כשמחשב מתקשר עם מחשב אחר, הוא צריך לדעת לאיזה "שירות" לפנות. **פורט** הוא מספר (0-65535) שמזהה את השירות. דמיינו בניין משרדים: כתובת ה-IP היא כתובת הבניין, והפורט הוא מספר החדר.

פורט	שירות	שימוש
80	HTTP	אתרי אינטרנט
443	HTTPS	אתרים מאובטחים
22	SSH	שליטה מרחוק (מסוכן!)
3389	RDP	שליטה במחשב Windows

מבנה הקבצים

```
log_analyzer/  
├─ main.py           # נקודת כניסה - מפעיל את התוכנית  
├─ config.py         # הגדרות (פורטים רגישים, ספים)  
├─ reader.py         # קריאת קובץ הלוג  
├─ checks.py         # פונקציות בדיקת חשדות  
└─ analyzer.py      # ניתוח וזיהוי חשודים
```

```
| reporter.py          # יצירת דוחות
| network_traffic.log  # קובץ הנתונים
```

דרישות עבודה עם Git

חשוב: כל פקודות Git יבוצעו בטרמינל בלבד - לא ב-GitKraken או כלים ויזואליים אחרים.

מבנה Branches: main (קוד יציב) + stage-1, stage-2 ... (ענף לכל שלב)

סדר עבודה:

```
git checkout -b stage-1      # חדש branch פתיחת
# ... כתיבת פונקציה ...
git add .
git commit -m "add load_csv function"
# ... כתיבת פונקציה נוספת ...
git commit -m "add filter_external_ips"
git push origin stage-1      # בסיום השלב
# main-ומיזוג ל-GitHub ב-PR פתיחת
```

מתי לעשות Commit: אחרי כל פונקציה שעובדת ונבדקה.

מתי לעשות PR: בסיום כל שלב, לפני מעבר לשלב הבא.

דרישות קוד

- חלוקה לפונקציות: כל משימה = פונקציה נפרדת. פונקציה עושה דבר אחד.
- שמות ברורים: filter_by_port ולא f1
- ללא קוד גלובלי: כל הקוד בתוך פונקציות, הפעלה דרך main().
- אבל - אחרי כתיבת כל פונקציה, בידקו אותה! כתבו קריאה זמנית לפונקציה, ודאו שהיא עובדת, ורק אז עשו commit ומחקו את קוד הבדיקה.