

Exam Structure- Tzach Aker

Section 1: Q&A (20 Questions)

• Terraform Fundamentals (5 questions)

1. Terraform הוא כלי לניהול תשתיות כמו שרתים ורשתות על ידי קוד, וכך קל לשחזר, לעדכן ולתחזק את התשתית כי הכל מתבצע באמצעות קוד. ההבדל העיקרי בין Terraform לכלים אחרים הוא ש Terraform לא מבצע פעולות ישירות, אלא יוצר קובץ קונפיגורציה ומבצע אותו בצורה אוטומטית.
2. Terraform הוא דקלרטיבי, הכוונה היא שאני רק אומר מה אני רוצה שיהיה, ולא איך להשיג את זה. State management שומר את המידע אודות המצב הנוכחי של התשתית, וכך Terraform יכול לדעת מה כבר קיים בתשתית ומה צריך לעדכן.
3. ה provider ב-Terraform הוא רכיב שמחבר את Terraform לשירותים חיצוניים כמו AWS ו-Azure ומבצע פעולות עליהם בקוד ה-terraform.
4. Terraform מזהה לבד בצורה אוטומטית את התלויות בין הרכיבים ומבצע אותם בסדר הזה שהתלויות יהיו קודם וכל רכיב ייווצר לאחר מכן.
5. קובץ קונפיגורציה של Terraform מכיל רכיבים כמו providers, resources, variables, outputs ועוד.

• State Management & Backend Configuration (3 questions)

6. Terraform refresh מעדכן את ה state של Terraform לפי המצב הנוכחי של התשתית. Terraform plan מציג את הפעולות ש Terraform הולך לבצע. Terraform apply מבצע את השינויים שהוצגו ב-plan.
7. Local Backend שומר את ה state על המחשב המקומי, ו remote backend שומר את ה state על שרת (כמו S3 ב-AWS). כשיש קבוצה שעובדת על התשתית מומלץ ה remote backend.
8. כדי להמנע מבעיות, אפשר להשתמש ב remote backend ששומר את ה state על השרת, ולוודא שכולם משתמשים באותו קובץ state משותף.

• Terraform Modules & Reusability (4 questions)

9. Terraform modules מאפשרים לשמור על תחזוקה טובה של התשתית על ידי זה שמאפשרים לשתף קוד ונמנעים משכפול של הקוד מספר פעמים.

10. משתמשים במילה `variable` בתוך `module`, מעבירים את הערכים כשקוראים ל-`module`.

11. `count` יוצר מספר מופעים בעזרת מספר מוגדר, ואילו `for_each` יוצר מופעים על פי רשימה או אובייקט.

12. אפשר להגדיר את ה-`source` של ה-`module` כ-URL של המאגר ב-`Git repository`. ניתן לרשום כך: `source=git::...."`

• Terraform with AWS (4 questions)

13. יוצרים `resource` מסוג `aws_instance` ומגדירים את השדות הנדרשים ל-`EC2` כמו סוג המחשב, `AMI` מפתח גישה וכו'.

14. בשביל להגדיר `VPC` צריך במינימום את השדות הבאים: `cidr_block` טווח כתובות ה-`IP` של ה-`VPC`, ו-`enable_dns_support`.

15. `Terraform` משתמש ב-`resource` מסוג `aws_iam_policy` כדי להגדיר מדיניות גישה ב-`AWS`.

16. משתמשים ב-`resource` מסוג `aws_lb` כדי להקים את ה-`Elastic Load Balancer` ואז מקשרים אותו עם `services` או `EC2 instances`.

17. `terraform validate` בודק אם קובץ הקונפיגורציה חוקי לפני ביצוע הפעולות השונות.

18. אפשר להשתמש בפקודות כמו `terraform plan` שבו רואים את הפעולות ש `terraform` יבצע ולקרוא את הודעות `errors` ולטפל בהם לפני ביצוע.

19. המדיניות של `ignore_changes lifecycle policy` ב-`Terraform` מונעת שינויים ברכיב מסוים כש `Terraform` משווה את ה-`state` עם המצב הנוכחי.

20. אפשר להשתמש ב- `terraform import` כדי להביא `AWS infrastructure` לתוך קובץ `state` של `Terraform`.