Backup and recovery (Ses. 3 & 4): Key takeaways

- Lab1: working with your database: in this Lab you learned to:
  - Connect to your Oracle Cloud Infrastructure subscription and create an Autonomous Database (either Autonomous Data Warehouse or Autonomous Transaction Processing)
  - Access to SqlDeveloper Web from ADB development Console
  - Create your own schema from SH sample schema: create table and load data from the SH sample schema
  - Publish your own schema as REST enabled: to allow querying your data through REST API calls.
  - Code a PL/SQL program that will support the POST method to create new records in your schema 's model
  - Create a JSON file and use the POST method to insert data in a table, from this JSON file

- Backup and Recovery:
  - RTO: Recovery Time Objective
  - RPO: Recovery Point Objective
  - Both are technical metrics, but are constrained by business requirements.
    - Hence must be defined by business in each business case
    - Costs grow when RTO and/or RPO tend to zero.
  - Backup & Recovery is the first step to MAA architecture (MAA stands for Maximum Availability Architecture). It's required to get a protection against user errors and media failures.
    - Logical backup: a mere "export" of the data stored in database's datafiles. Is usually associated with huge values of RTO and RPO.
    - Physical backup: a copy of database's files (datafiles, dictionary, undo, redo)
      - **Cold backup**: copy is made when database has been cleanly shutdown, hence the backup is **consistent** (doesn't need recovery after restoring). Implies a business discontinuity. RTO is typically the time needed to restore the file, from backup location to database location. RPO can be high, typically the time between the crash and the last available backup.
      - **Hot backup**: copy is made when database is up and running, servicing client applications. Hence hot backup are **always inconsistent**, meaning that a recovery will be necessary after restoring. RTO is higher than with cold backup (need to recover), RPO can be minimized near zero, as we recover until the moment of the crash applying REDO from archived redologs and UNDO.
  - Apply best practices to meet RTO and RPO, depending of their expected values (business requirements).
  - Always test your policy, whatever simple it might be. Backup, restore and recover must be carefully tested, and RTO/RPO must be measured during the tests, to check they meet business requirements in any case.

- A typical backup policy: hot full backup once a week (Sunday night), incremental backup and archived redologs backup the rest of the days (by night). With such a policy, if you have to recover from a complete database crash that occurred on Thursday, you will hopefully:
  - Restore Sunday's full backup
  - Apply incremental backups until Wednesday
  - Recover with archived redologs until Thursday