1. **Implicit vs Explicit wait** –  
   Implicit wait kehtib kõigile elementidele ja paneb WebDriveri ootama, kuni element ilmub.  
   Explicit wait on konkreetse olukorra jaoks (nt oodatakse, kuni element muutub nähtavaks).  
   Tavaliselt kasutatakse explicit wait’i, sest see on täpsem. Neid kahte ei soovitata koos kasutada – tekitavad liigseid viivitusi.
2. **findElement vs findElements** –  
   findElement() leiab ühe elemendi ja viskab vea (NoSuchElementException), kui seda pole.  
   findElements() tagastab nimekirja ja kui midagi ei leitud, annab tühja listi.
3. **NoSuchElementException vs TimeoutException** –  
   NoSuchElementException tähendab, et elementi pole üldse olemas.  
   TimeoutException tuleb siis, kui oodatud tingimus (nähtavus vms) ei täitunud kindla aja jooksul.
4. **StaleElementReferenceException** –  
   Juhtub siis, kui element on vahepeal DOM-ist muutunud või kadunud (nt leht uuesti laetud).  
   Lahendus: otsi element uuesti või kasuta explicit wait’i.
5. **CSS vs XPath** –  
   CSS on kiirem ja lihtsam lugeda, sobib enamiku juhtudega.  
   XPath on võimsam, kuid aeglasem ja keerulisem.  
   Seetõttu eelistatakse CSS-i umbes 80% juhtudest.
6. **Page Object Model (POM)** –  
   Iga leht on eraldi klass, kus on locatori ja meetodid.  
   Testikood ei pea teadma, kuidas element leitakse.  
   Kui midagi muutub, muudad ainult Page klassis – testid jäävad samaks.
7. **Headless Chrome** –  
   Mõnikord käitub erinevalt kui tavaline brauser (nt stiilid, pildid, hover-efektid).  
   Probleemid tekivad, sest graafiline osa puudub – tuleb testida mõlemas režiimis.
8. **Implicit + Explicit koos** –  
   Kui mõlemad on määratud, võivad ootamised kattuda ja test aeglustub.  
   Soovitatav on kasutada ainult explicit wait’i.
9. **Chrome avatuks jätmine (detach)** –  
   Lisa chrome\_options.add\_experimental\_option("detach", True) – nii jääb aken avatuks pärast testi.  
   Kui ei tööta, eemalda driver.quit().
10. **== vs .equals() Java-s** –  
    == võrdleb viiteid (kas mõlemad osutavad samale objektile).  
    .equals() võrdleb väärtust.  
    Stringide puhul kasuta alati .equals().
11. **List vs ArrayList** –  
    List on liides, ArrayList konkreetne klass.  
    Kui kasutad muutujana List, saad hiljem lihtsalt muuta teostust (nt LinkedList).
12. **assertEquals vs assertTrue (JUnit)** –  
    assertEquals võrdleb kahte väärtust ja annab selge vea-sõnumi.  
    assertTrue kontrollib lihtsalt tingimust, ilma detailsuseta.
13. **visibilityOfElementLocated vs presenceOfElementLocated** –  
    presence... ootab, et element oleks DOM-is.  
    visibility... ootab, et element oleks DOM-is ja nähtav.
14. **Maven Surefire: üksik test** –  
    Käivita klass: mvn test -Dtest=KlassiNimi  
    Käivita konkreetne meetod: mvn test -Dtest=KlassiNimi#meetodNimi
15. ***data- atribuudid*\* –**  
    data-test või data-id on loodud testide jaoks.  
    Need ei muutu koos kujundusega, seega locatorid on stabiilsemad.