Software Engineering Overview

1. Software engineering is an application of computer science principles and foundational knowledge. While computer science is more research, software engineers create solutions for end users.
2. Programming in the Small refers to smaller projects or tasks that are coded for a smaller purpose. It is dissimilar from programming in the Large as programming in the Large focuses on bigger picture products that could take years to complete and will likely be sold as a product or used for many years by the company that created it. programming in the Small does not take as much maintenance or upkeep
3. Greenfield development refers to the process of creating and coding something from a blank slate. Something new that you start from the ground up. Brownfield development is more common and refers to working on something that has already been worked on and developed possibly by another team or person. This involves dealing with legacy code and adding new features to already developed code. Greenfield development is risky as there is no guarantee or backing behind the project.
4. Four steps of maintenance
    1. Corrective: Deals with fixing flaws in the code or system. This can come from faults in the logic that affects the system after it has been released. Can cause other errors because of one specific error. Can also be referred to as bug fixing
    2. Perfective: Improving or perfecting the software. This can be improvements to the interface or functionality, or optimization. This does not deal with defects or errors.
    3. Adaptive: This involves fixing the software before it is deployed. If the software the code is running on is being updated and something would go wrong at launch, you would correct the code before the release. If you want to launch the software on a different platform to be compatible with that.
    4. Preventative: Something you might do to make sure errors don't happen in the future. Optimizing code just for the sake of it being better for the future, not particularly because of any complaints.
5. .
    1. Idea / Planning: Initiation, system concept development, planning
    2. Requirements: Requirements analysis
    3. Design: Design
    4. Coding: Development
    5. Testing: Integration and test,
    6. Deploy: Implementation

7. Maintenance: Operations and maintenance, disposition