Week 3 Quiz 2

1. How can an LLM assist with managing dependency conflicts?

   - By automatically resolving dependency conflicts.
   - By writing code without dependencies.
   - By helping convert transitive dependencies into direct dependencies
   - By recommending solutions to dependency conflicts

   An LLM can provide recommendations for solutions to resolving dependency conflicts

2. What should you do if an LLM does not provide a solution for a dependency conflict?

   - Ignore the conflict and continue development.
   - Continue prompting the LLM until it generates a solution
   - Conclude no solution exists and remove the dependencies causing the conflict
   - Look for alternative libraries or use traditional search methods for solutions.

   If an LLM does not provide a solution, looking for alternative libraries or using traditional search methods (e.g., Stack Overflow) can help resolve the dependency conflict.

3. How can LLMs assist with managing security in dependencies?

   - By identifying and providing information about vulnerabilities
   - By writing new dependencies
   - By replacing dedicated cybersecurity staff on your project
   - By removing dependencies from your project

   LLMs can help identify security vulnerabilities in dependencies and provide detailed information and recommendations for fixing them.

4. What should you do if you find an unmaintained package in your project?

   - Prioritize using this package as a lack of maintenance indicates a high level of confidence in its security
   - Gather input from an LLM, cybersecurity-focused colleagues, and other trusted sources as you consider replacing it
   - Delete the entire project as it is already likely compromised
   - Ignore the package as unmaintained packages can't be used as attack vectors

   If you find an unmaintained package, it's important to gather information about it using tools like ChatGPT and consider replacing it with a maintained and secure alternative.

5. Why are LLMs potentially imperfect tools for addressing dependency security?

    - LLMS cannot identify potential vulnerabilities unless explicitly prompted on them
    - LLMs may not have information on recently discovered security issues or very obscure libraries in their training data
    - LLMs are trained on security issues but not their resolutions
    - LLMs have not been trained on secure code and so are unable to produce it.

    LLMs are only as good as their training data. If a vulnerability wasn't known prior to the cutoff date on the training data for the model, or if it can't otherwise access information on that vulnerability, for example with a web search, they can't provide information on those vulnerabilities.