Part 3: Critical Thinking (20 points)
**Ethics & Bias (10 points)**

Q1: *How might biased training data affect patient outcomes in the case study?*

Biased training data can lead to inaccurate or unfair predictions, especially if certain patient groups (e.g., based on age, ethnicity, or socioeconomic status) are underrepresented or misrepresented. For example, if the dataset predominantly contains data from urban patients, the model may perform poorly on rural patients, potentially underestimating their risk of readmission. This could result in inadequate care or follow-up, widening healthcare disparities and causing ethical harm.

Q2: *Suggest 1 strategy to mitigate this bias.*

One effective strategy is data balancing through re-sampling or synthetic data generation (e.g., SMOTE) to ensure all patient subgroups are well-represented. Additionally, conducting fairness audits and performance evaluations across different demographic groups can help identify and correct biases before deployment.

**Trade-offs (10 points)**

Q3: *Discuss the trade-off between model interpretability and accuracy in healthcare.*

In healthcare, interpretability is often as important as accuracy. Highly accurate models like deep neural networks may offer powerful predictions, but their 'black-box' nature makes it difficult for doctors and regulators to understand or trust the output. On the other hand, simpler models like decision trees or logistic regression may have slightly lower accuracy but offer clear insights into decision-making logic. In clinical settings, interpretable models are usually preferred to ensure transparency, accountability, and compliance.

Q4: *If the hospital has limited computational resources, how might this impact model choice?*

Limited computational resources can restrict the hospital from using complex, resource-intensive models like deep learning. Instead, the hospital may opt for lightweight, efficient algorithms such as logistic regression, decision trees, or Random Forests with limited depth. These models train faster, require less memory and processing power, and are easier to maintain on local servers or embedded systems.