1. [2 points] How can a DBMS provide data independence?

Anything that makes sense will do it here. For example: DBMS provides an abstract view of the data (external schema and conceptual schema) to the application programs that hides the details of data representation and storage. Check 1.5.3 from the textbook.

2. Suppose that we have a database with the following schema:

   - **Product** (maker: string, model: int, type: string)
   - **PC** (model: int, speed: int, ram: int, hd: int, price: int)
   - **Laptop** (model: int, speed: int, ram: int, hd: int, screen: string, price: int)
   - **Printer** (model: int, color: boolean, type: string, price: int)

Write expressions of relational algebra to answer the following queries.

   a) [2 points] What PC models have a speed of at least 3.00?

   \[
   R1 := \sigma_{\text{speed} \geq 3.00} (PC) \\
   R2 := \pi_{\text{model}} (R1)
   \]

   b) [2 points] Which manufacturers make laptops with hard disk of at least 100GB?

   \[
   R1 := \sigma_{\text{hd} \geq 100} (Laptop) \\
   R2 := \text{Product} \bowtie (R1) \\
   R3 := \pi_{\text{maker}} (R2)
   \]

   c) [2 points] Find the model number of all color laser printers

   \[
   R1 := \sigma_{\text{color} = \text{true} \ \text{AND} \ \text{type} = \text{laser}} (Printer) \\
   R2 := \pi_{\text{model}} (R1)
   \]
d) [2 points] Find those hard disk sizes that occur in two or more PCs

\[
\begin{align*}
R1 & := \rho(PC1,PC) \\
R2 & := \rho(PC2, PC) \\
R3 & := R1 \bowtie (PC1.hd = PC2.hd \text{ AND } PC1.model \neq PC2.model) R2 \\
R4 & := \pi_{hd}(R3)
\end{align*}
\]

e) [3 points] Find the manufacturers of the PC or Laptop with the highest available speed.

\[
\begin{align*}
R1 & := \pi_{model,speed}(PC) \\
R2 & := \pi_{model,speed}(Laptop) \\
R3 & := R1 \cup R2 \\
\rho(R4(model\rightarrow model2, speed\rightarrow speed2), R3) \\
R5 & := \pi_{model,speed}(R3 \bowtie (speed < speed2) R4) \\
R6 & := R3 – R5 \\
R7 & := \pi_{maker}(R6 \bowtimes Product)
\end{align*}
\]

3. [7 points] Draw the ER diagram for a bank database. The design shall include information about customers and their accounts. Information about customers includes their name, address (street, city, and state), phone (area code and number), and SSN. Accounts have numbers, type (i.e. saving or checking), and balances. A customer can own one or more accounts. Also a customer can have one or more addresses and one or more phone numbers. And of course a phone or address belong only to a single customer. An account can be owned by one customer only. Remember that we assume attributes are of primitive type, that is, you can NOT have an attribute as a set, structure, array, etc. You need to decide on the primary key for each entity set.