Q1) The basic issue with the current table is that it violates the first normal form (1NF) by having repeating groups in the "Movie Rented" column. Each cell in this column contains multiple values indicating that a customer can rent multiple movies. To resolve this, we need to create a separate row for each customer-movie combination.

Here's how the new table looks after converting it to 1NF:

customer_name	title	Address	Movie Rented
Lokesh Daga	Mr	403-B Marwadi nagar.	Race 2
Lokesh Daga	Mr	403-B Marwadi nagar	Radhe
Lokesh Daga	Mr	403-B Marwadi nagar	Bharat
Meelabh Shukla	Mr	419-B1 Kota nagar	Daddy's Little Girls
Lokesh Daga	Mr	57 Ashok Nagar	The Notebook.
Rashi Sharma	Ms	109 Ram nagar. Napasar	Fanna
Rashi Sharma	Ms	109 Ram nagar Napasar	The Notebook

In the new table, each row represents a single movie rented by a customer. The repeating groups in the "Movie Rented" column have been eliminated and the customers name, title, address, and movie rented are stored as separate attributes.

Q2 The keys that uniquely identify the tuples in the table are the combination of the following attributes.

① customer_name
② title
③ Address.

A combination of these attributes uniquely identifies each row in the table.

Q3) To convert the table into its second normal form (2NF), we need to eliminate partial dependencies. In the current table, the "Address" column is functionally dependent on both the customer_name and the attribute However, the "Address" attribute is actually dependent only on the custom -er_name.

Here are the new tables in 2NF:

Table1: Customers.

Customer_name	title.
Lokesh Daga	Mr.
Neelakh Shukla	Mr
Rashi Sharma	Ms.

Table2: Rentals

Customer_name	Movie_Rented	Address
Lokesh Daga	Race2	403-B Marwadi Nagar
Lokesh Daga	Radhe	403-B Marwadi Nagar
Lokesh Daga	Bharat	403-B Marwadi Nagar
Neelabh Shukla	Daddy's Little Girls	403-A 49-M Kota nagar
Lokesh Daga	The Notebook	57 Ashok Nagar
Rashi Sharma	Panna	109 Ram nagar. Napasar
Rashi Sharma	The Notebook	109 Ram nagar Napasar.

In the new tables, the first table "Customers" contains unique customer names and titles. The second table "Rentals" contains the rental information, with customer_name and title as foreign keys referencing the "Customers" table. The "Address" attributes is stored in the "Rentals" table, eliminating the partial dependency.

04) Yes, we can add new movies like "The Jungle Book" and "Fast and Furious 9" to the database, even if they are not yet rented to anyone. We can simply insert new rows into the "Rentals" tables with the respective movie title and leave the computer-name and address attributes as NULL or empty.

Table 2: Rentals (after adding new movies)

Customer name	Movie rented	Address
Lokesh Daga	Race2	403-B Marwadi Nagar
Lokesh Daga	Radhe	403-B Marwadi Nagar
Lokesh Daga	Bharat	403-B Marwadi Nagar
Neelabh Shukla	Daddy's Little Girls.	419-M Kota Nagar.
Lokesh Daga	The Notebook.	57 Ashok Nagar.
Rashi Sharma	Fanna	109 Ram nagar. Napasar.
Rashi Sharma	The Notebook	109 Ram Nagar Napasar.
NULL	The Jungle Book.	NULL
NULL	Fast and Furious 9	NULL.

05) The tables in 2NF are already in a normalized form. There are no partial dependencies or non-key attributes dependent on a subset of a candidate key. Therefore, there is no need to further convert these tables to a higher normal form.

06) Since the table are already in their highest normal form (2NF), there is no need to convert them into a higher normal form. The tables are properly structured and eliminated all forms of functional dependencies