Only use SELECT statements to provide the specific output requested (no temporary tables or functions).
All months stated are in the year 2018.

1. Find the number of joins per day for March.
2. Find the number of unique members that joined per day for March.
3. Find the product (ID #) that brought in the most revenue in March.
4. Find the product (ID #) that brought in the second most revenue in March.
5. Find the average revenue per member that signed up in January within 1 month and 2 months of their signup date.

Table 1 – Memberships

member_id (primary key)	int	NOT NULL	
signup_date	datetime	NOT NULL	Date of first join
join_country	varchar(2)	NULL	Can be null if using proxy
email	varchar	NOT NULL	
cancellation_date	date	NULL	Can be null if member is still active or failed payment
expiry_date	date	NULL	Can be null if member is still active

Table 2 - Transactions

transaction_id (primary key)	int	NOT NULL	
member_id	int	NOT NULL	
transaction_date	datetime	NOT NULL	
transaction_type_id	int	NOT NULL	1 if join, 2 if rebill
product_id	int	NOT NULL	
transaction_amount	float	NOT NULL	

Table 3 - Logins

login_id (primary key)	int	NOT NULL	
member_id	int	NOT NULL	
product_id	int	NOT NULL	
login_date	datetime	NOT NULL	
login_country	Varchar(2)	NULL	Can be null if using proxy

Table 4 - Searches

search_id (primary key)	int	NOT NULL	
member_id	int	NULL	Can be null if user has not joined or logged in
product_id	int	NOT NULL	
search_date	datetime	NOT NULL	
search_term	varchar	NOT NULL	