```
import json


from pydantic import BaseModel




def base_model_to_json(model: BaseModel, indent: int = 3):
    """
    Converts the JSON schema of a base model to a formatted JSON string.


    Args:
        model (BaseModel): The base model for which to generate the JSON schema.


    Returns:
        str: The JSON schema of the base model as a formatted JSON string.
    """
    out = model.model_json_schema()
    return str_to_json(out, indent=indent)




def extract_json_from_str(response: str):
    """
    Extracts a JSON object from a string.


    Args:
        response (str): The string containing the JSON object.
    Returns:
        dict: The extracted JSON object.

    Raises:
        ValueError: If the string does not contain a valid JSON object.
    """
    json_start = response.index("{")
    json_end = response.rfind("}")
    return json.loads(response[json_start : json_end + 1])


def str_to_json(response: str, indent: int = 3):
    """
    Converts a string representation of JSON to a JSON object.

    Args:
        response (str): The string representation of JSON.
         indent (int, optional): The number of spaces to use for indentation in the JSON output. Defaults
to 3.

    Returns:
        str: The JSON object as a string.

    """
    return json.dumps(response, indent=indent)
```