const getContenidoById = async (req, res) => {

  try {

    const contenidoID = req.params.id;

    const contenido = await Contenido.findByPk(contenidoID);

    contenido

      ? res.status(200).json(contenido)

      : res.status(404).json({ error: "Contenido no encontrado" });

  } catch (error) {

    console.error(error);

    res.status(500).json({ message: "Error al obtener Contenido" });

  }

};

const createContenido = async (req, res) => {

  try {

    const { poster, titulo, resumen, temporadas, trailer, categoria\_id, genero\_id } = req.body;

    const nuevoContenido = await Contenido.create({

      poster,

      titulo,

      resumen,

      temporadas,

      trailer,

      categoria\_id,

      genero\_id,

    });

    res.status(201).json(nuevoContenido);

  } catch (error) {

    console.error(error);

    res.status(500).json({ message: "Error al crear Contenido" });

  }

};

const updateContenido = async (req, res) => {

  try {

    const { id } = req.params;

    const { poster, titulo, resumen, temporadas, trailer, categoria\_id, genero\_id } = req.body;

    const [updated] = await Contenido.update({

      poster,

      titulo,

      resumen,

      temporadas,

      trailer,

      categoria\_id,

      genero\_id,

    }, {

      where: { id },

    });

    if (updated) {

      const updatedContenido = await Contenido.findByPk(id);

      res.status(200).json(updatedContenido);

    } else {

      res.status(404).json({ error: "Contenido no encontrado" });

    }

  } catch (error) {

    console.error(error);

    res.status(500).json({ message: "Error al actualizar Contenido" });

  }

};

const deleteContenido = async (req, res) => {

  try {

    const { id } = req.params;

    const deleted = await Contenido.destroy({

      where: { id },

    });

    deleted

      ? res.status(204).json()

      : res.status(404).json({ error: "Contenido no encontrado" });

  } catch (error) {

    console.error(error);

    res.status(500).json({ message: "Error al eliminar Contenido" });

  }

};