```
await self.delall_active(group_id)
            await self.delall_filters(group_id)
            await self.del_main(group_id)
            await self.refresh_cache(group_id)

        return


    async def del_main(self, group_id: int):
        """
        A Funtion To Delete the chat's main db document
        """
        await self.col.delete_one({"_id": group_id})
        await self.refresh_cache(group_id)

        return True


    async def refresh_cache(self, group_id: int):
        """
        A Funtion to refresh a chat's chase data
        in case of update in db
        """
        if self.cache.get(str(group_id)):
            self.cache.pop(str(group_id))

        prev = await self.col.find_one({"_id": group_id})

        if prev:
            self.cache[str(group_id)] = prev
        return True

    # Related To Finding Active Channel(s)
    async def add_active(self, group_id: int, channel_id: int, channel_name):
        """
        A Funtion to add a channel as an active chat the a connected group
        (This Funtion will be used only if its the first time)
        """
        templ = {"_id": group_id, "chats":[{"chat_id": channel_id, "chat_name":
channel_name}]}

        try:
            await self.acol.insert_one(templ)
            await self.refresh_acache(group_id)
        except Exception as e:
            print(e)
            return False

        return True



    async def del_active(self, group_id: int, channel_id: int):
        """
        A funtion to delete a channel from active chat colletion in db
        """
        templ = {"$pull": {"chats": dict(chat_id = channel_id)}}

        try:
            await self.acol.update_one({"_id": group_id}, templ, False, True)
        except Exception as e:
            print(e)
            pass

        await self.refresh_acache(group_id)
        return True


    async def update_active(self, group_id: int, channel_id: int, channel_name):
        """
        A Funtion to add a new active chat to the connected group
        """
        group_id, channel_id = int(group_id), int(channel_id)

        prev = await self.acol.find_one({"_id": group_id})
        templ = {"$push" : {"chats" : dict(chat_id = channel_id, chat_name = 
channel_name)}}
        in_c = await self.in_active(group_id, channel_id)

        if prev:
            if not in_c:
                await self.acol.update_one({"_id": group_id}, templ)
            else:
                return False
        else:
            await self.add_active(group_id, channel_id, channel_name)
        return True


    async def find_active(self, group_id: int):
        """
        A Funtion to find all active chats of
        a group from db
        """
        if self.acache.get(str(group_id)):
            self.acache.get(str(group_id))

        connection = await self.acol.find_one({"_id": group_id})

        if connection:
            self.acache[str(group_id)] = connection
            return connection
        return False


    async def in_active(self, group_id: int, channel_id: int):
        """
        A Funtion to check if a chat id is in the active
        chat id list in db
        """
        prev = await self.acol.find_one({"_id": group_id})

        if prev:
            for x in prev["chats"]:
                if x["chat_id"] == channel_id:
                    return True

            return False
        return False


    async def delall_active(self, group_id: int):
        """
        A Funtion to Delete all active chats of
        a group from db
        """
```