

分区

```
...
-- DROP FUNCTION public.create_data_collection_partition_notice();

CREATE OR REPLACE FUNCTION
public.create_data_collection_partition_notice()
RETURNS trigger
LANGUAGE plpgsql
AS $function$
DECLARE
    childNum int;
BEGIN
    SELECT count(child.relname) FROM pg_inherits JOIN pg_class parent
        ON pg_inherits.inhparent = parent.oid JOIN pg_class child
        ON pg_inherits.inhrelid = child.oid JOIN pg_namespace
nmsp_parent
        ON nmsp_parent.oid = parent.relnamespace JOIN pg_namespace
nmsp_child
        ON nmsp_child.oid = child.relnamespace WHERE parent.relname =
't_data_collection' INTO childNum;
    IF NEW.collection_id > (childNum-1)*10000000 AND (SELECT EXISTS
(SELECT FROM pg_tables WHERE tablename = FORMAT('%s_%s',
TG_ARGV[0], childNum))) = false
    THEN
        EXECUTE
            format('NOTIFY t_data_collection, ''%s'', childNum);
    END IF;
    RETURN NULL;
END;
$function$
;
...

### create\_table\_partition(table\_name)

...
-- DROP FUNCTION public.create_table_partition(varchar);

CREATE OR REPLACE FUNCTION
```

```

public.create_table_partition(table_name character varying)
    RETURNS text
    LANGUAGE plpgsql
AS $function$
    DECLARE
        childNum int;
    BEGIN
        SELECT count(child.relname) FROM pg_inherits JOIN pg_class
parent
        O stener;
import lombok.extern.slf4j.Slf4j;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.scheduling.annotation.EnableAsync;
import org.springframework.scheduling.annotation.Scheduled;
import org.springframework.stereotype.Component;

import javax.sql.DataSource;
import java.sql.SQLException;

/**
 * @author zhuye
 * @version 1.0
 * @description: TODO
 * @date 2023/11/8 10:44
 */
@Component
@Slf4j
@EnableAsync
public class PGListenTimer {

    @Autowired
    private DataSource dataSource;

    private PGNoticeListener dataCollectionListener;

    private PGNoticeListener pageDataValueListener;

    /**
     * 定时检查PGListener是否正常
     * initialDelay: 容器启动后多长时间第一次执行
     * fixedDelay: 间隔时间 10seconds
     */
    @Scheduled(initialDelay = 2000, fixedDelay = 10000)
    public void dataCollectionListener() {
        if (BeanUtil.isEmpty(dataCollectionListener) ||
!dataCollectionListener.isAlive()) {
            try {
                dataCollectionListener = new

```

```
PGNoticeListener(dataSource.getConnection(), "t_data_collection");
    dataCollectionListener.start();
} catch (SQLException e) {
    throw new RuntimeException(e);
}
}
}
}
}
...
```

原文链接: <https://juejin.cn/post/7387263949250314276>