



<http://www.lycato.com>



Spring Boot

Scheduling Tasks

Nguyễn Nghiệm





- ☐ SCHEDULED TASKS
- ☐ BUILDING BACKGROUND SERVICES

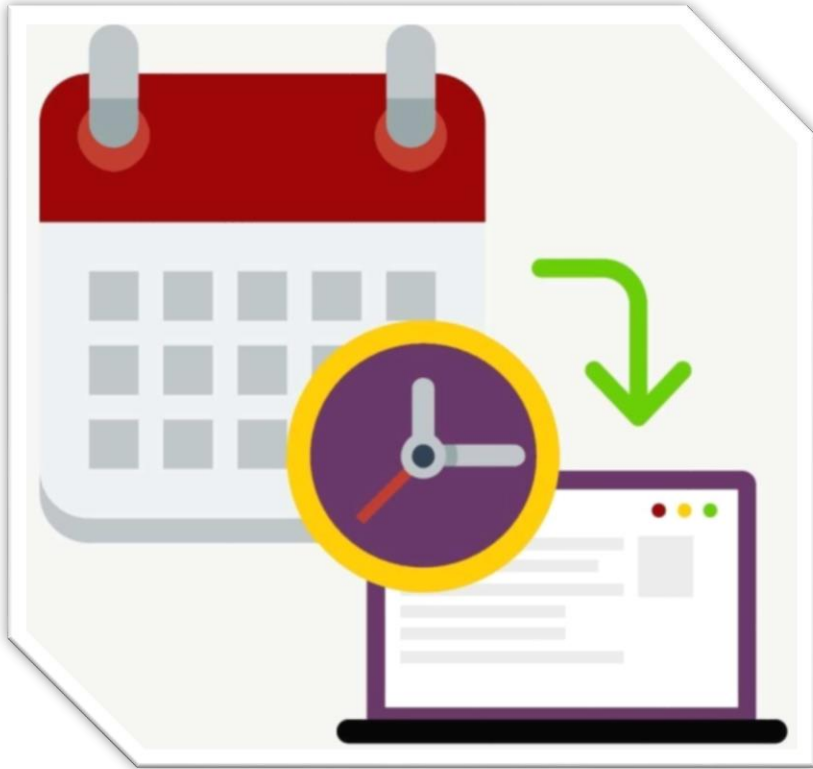




Scheduling Tasks



SCHEDULED TASK



```
@Service
public class BackgroundService{
    @Scheduled(fixedRate = 1000)
    public void run() {
        System.out.println(new Date());
    }
}
```

2

Thiết lập lịch trình

```
@SpringBootApplication
@EnableScheduling
public class MainApplication {...}
```

1

Kích hoạt ứng dụng



❑ *@Scheduled*(*fixedRate*, *fixedDelay*, *cron*, *initialDelay*)

❖ *initialDelay* = 5000

➤ Bắt đầu chạy lần đầu tiên sau khi ứng dụng start 5 giây

❖ *fixedRate* = 1000 // *mili giây*

➤ Mỗi giây thực hiện một lần

❖ *fixedDelay* = 1000 // *mili giây*

➤ 1 giây sau khi thực hiện xong sẽ thực hiện lần kế tiếp

❖ *cron* = "0 * * * * MON-FRI" // "*second minute hour date month weekday*"

➤ Phút đầu tiên các ngày từ thứ 2 đến thứ 6 sẽ thực hiện

✓ MON-FRI: liên tục, MON,FRI: rời rạc

❑ Chỉ sử dụng 1 trong 3 *fixedRate*, *fixedDelay*, *cron*

❑ Scheduled method: *không tham số và không trả về*

❖ *void method()*



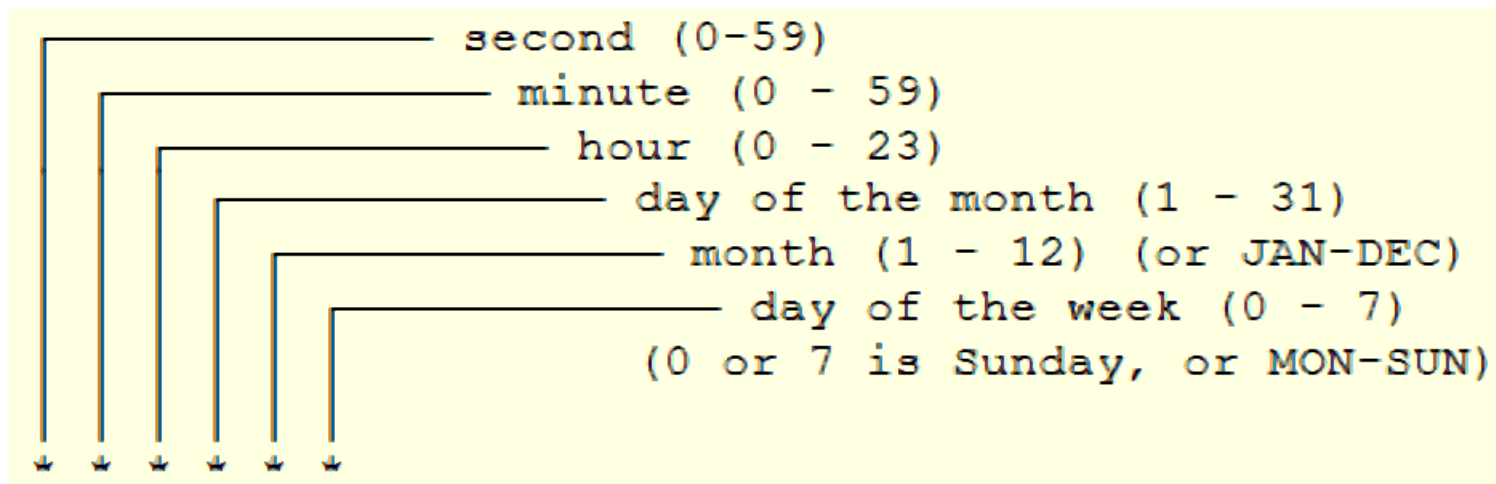
```
@Scheduled(  
void method() { ... }
```



@SCHEDULED(FIXEDRATE, FIXEDDELAY, CRON, INITIALDETAY)



@Scheduled(cron, initialDelay)



cron="0 * * * * MON-FRI"

-> Giấy thứ nhất các ngày trong tuần

cron="59 59 23 1 * *"

-> Giấy cuối cùng các ngày đầu mỗi tháng

cron="* * 1,13 * * *"

-> 1 giờ và 13 giờ hằng ngày

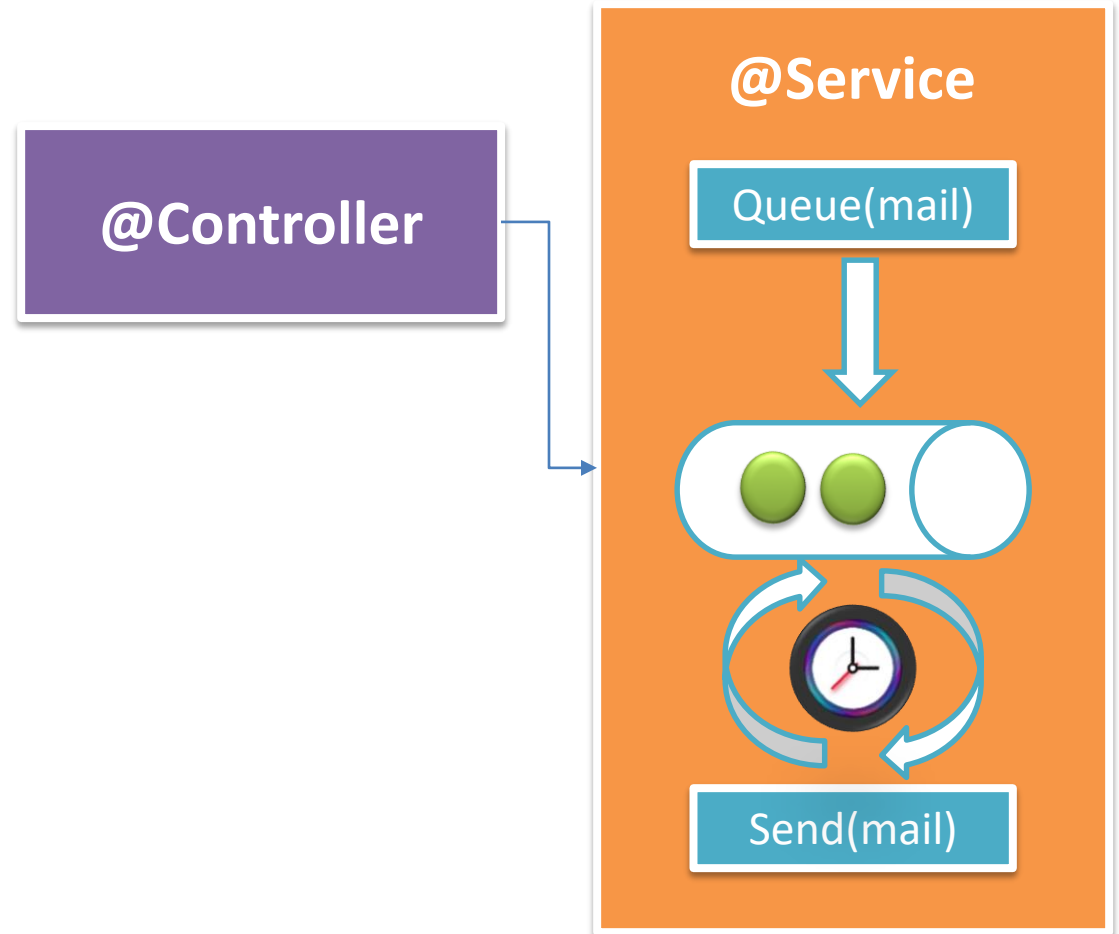


Building Background Service



ỨNG DỤNG SCHEDULED TASK

- ☐ Gửi email theo định kỳ, chiến dịch
- ☐ Xóa dữ liệu rác
- ☐ Vô hiệu hóa dữ liệu hết hiệu lực
- ☐ Sao lưu dữ liệu...
- ☐ Xử lý hàng đợi các công việc chiếm nhiều thời gian như gửi email




```
public interface MailerService{  
    void queue(Mail mail);  
}
```

```
@Controller  
public class MailerController{  
    @Autowired  
    MailerService mailer;  
    ...  
    mailer.queue(mail);  
}
```

@Service

```
public class MailerServiceImpl implements MailerService{  
    List<Mail> mails = new ArrayList<>();  
    @Override  
    public void queue(Mail mail) {  
        mails.add(mail);  
    }  
}
```

```
@Scheduled(fixedDelay = 2000)  
public void run() {  
    while(!mails.isEmpty()) {  
        Mail mail = mails.remove(0);  
        this.send(mail);  
    }  
}  
private void send(Mail mail) {...}
```

```
}
```



✓ @ENABLESCHEDULING

✓ @SCHEDULED

✓ FIXEDRATE

✓ FIXEDDELAY

✓ CRON

✓ INITDELAY

✓ BUILDING MAIL SENDER IN BACKGROUND





Thank
you