



INSTITUTO FEDERAL
PARANÁ

DESENVOLVIMENTO PARA DISPOSITIVOS MÓVEIS

PROF^a. M.Sc. JULIANA H Q BENACCHIO



Utilização de log

- Utilizar *logs* é uma forma de mostrar informações úteis durante a execução do programa
- A forma mais simples de fazer isso no Java é através de **`System.out.println()`**
 - Não é possível categorizar o erro



Utilização de log

- O **LogCat** é um mecanismo versátil de *logging* do Android
- A classe **android.util.Log** possui alguns métodos utilitários para imprimir informações com os níveis de detalhes desejados.
- Os métodos da classe **Log** são bem simples, e começam com a primeira letra do nível de severidade.



- Por exemplo, para imprimir um **log** com nível de alerta é utilizado o método

Log.w(tag, mensagem)

| Método | Tipo de Mensagem |
|---------|-------------------------------|
| Log.v() | Verbose (depuração detalhada) |
| Log.d() | Debug (depuração) |
| Log.i() | Info (informação) |
| Log.w() | Warning (aviso) |
| Log.e() | Error (erro) |



Utilização de log

- Cada **log** pode ser escrito em uma determinada TAG, para que posteriormente apenas as mensagens desejadas sejam recuperadas.

```
Log.i("Tela Inicial", "Iniciando o processo");
```

Log de mensagem
de informação

A tag permite
categorizar a mensagem



Utilização de log

- A constante **TAG** pode ser definida na classe e utilizada como identificador das mensagens.
- Isso é útil para filtrar as mensagens por tags/categorias, facilitando a visualização e debug.
- Os logs gerados podem ser visualizados na janela **LogCat**.



Utilização de log



```
...
import android.util.Log;

public class DebugActivity extends Activity {

    private static final String TAG = "TADS";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_debug);
        ...
        Log.v(TAG, "log de verbose");
        Log.d(TAG, "log de debug");
        Log.i(TAG, "log de info");
        Log.w(TAG, "log de alerta");
        Log.e(TAG, "log de erro");
    }
    ...
}
```



- Acostume-se a utilizar o **LogCat**, pois é nessa janela que todas as mensagens de *logs* e erros são exibidas.
- Em caso de erro, o “*stack trace*” das exceções aparece no **LogCat** e é possível verificar o número da linha que originou o problema.
- Sempre que o seu aplicativo travar (**force close**), olhe as mensagens de erro no **LogCat**.

